MOTOROLA

MPAA3UM/D

EasyAnalog™ DESIGN SOFTWARE
User's Manual

MPA

Comprehensive Instructions for Using Motorola's EasyAnalog Design Software to Configure Motorola Field Programmable Analog Arrays

MOTOROLA

EasyAnalog™ Design Software

User's Manual

Table of Contents

Introduction

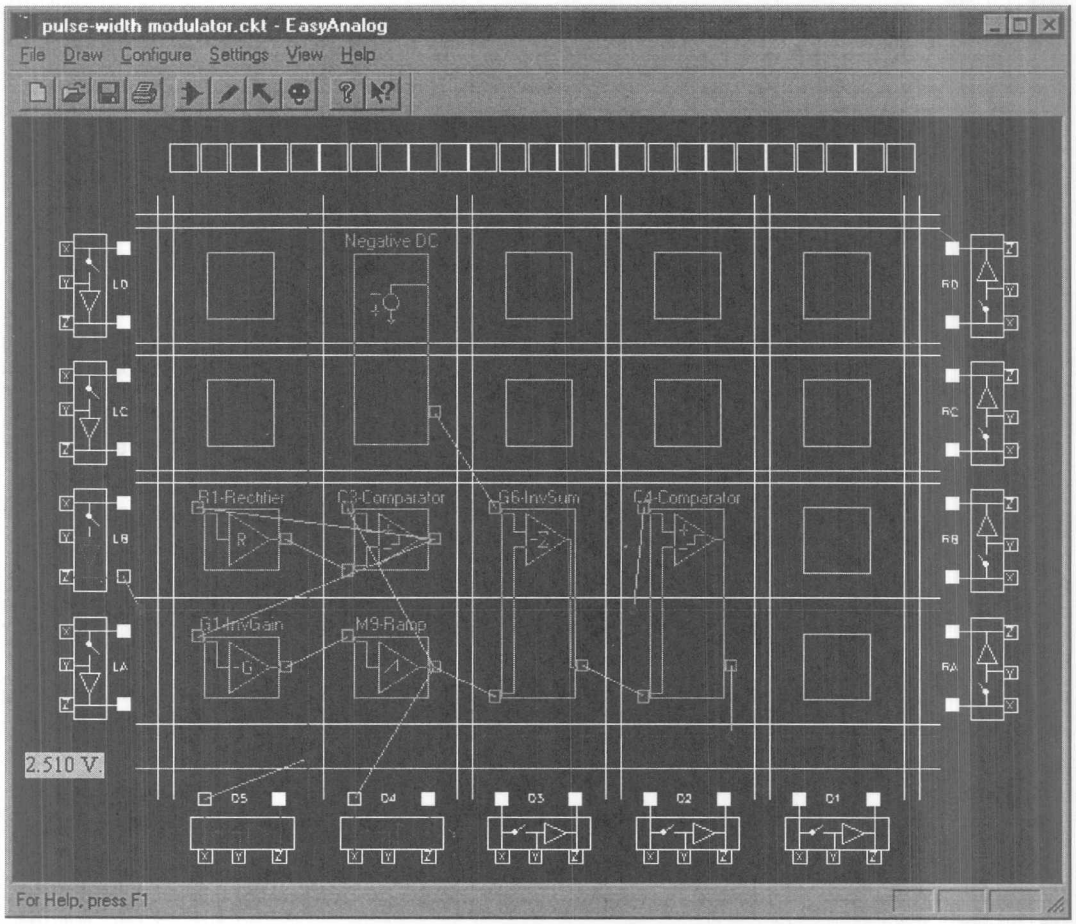
What does this software do?

EasyAnalog software allows you to quickly and easily construct complex analog circuits by selecting, placing and wiring building-block circuits referred to as *Macros (see page 60)*. The analog circuits that you build may then be down-loaded to Motorola's new Field Programmable Analog Array (FPAA) chip via the PC computer's serial port. The chip will then begin functioning as the circuit you constructed. You can see the results of your analog design immediately using a signal generator and an oscilloscope.

The chip can be reprogrammed as many times as desired so you can try out as many circuits as you like.

Also note that multiple, independent circuits can be constructed and run simultaneously, each with its own independent inputs and outputs. For example, you could have 3 gain stages, 2 rectifiers and a comparator operating totally independently and simultaneously each with its own input and output terminals.

Shown below is one of the thousands of possible circuits that can be created using EasyAnalog software and the Motorola Field Programmable Analog Array chip.



EasyAnalog software Display of a Pulse-Width Modulator

How does the chip work?

Motorola's Field Programmable Analog array is based on switched-capacitor technology, i.e., a capacitor can be made to function as a resistor if you repeatedly switch it in and out of a circuit. Using switched-capacitors and opamps, elemental analog amplifiers can be created whose gain can be controlled by selecting the values of the capacitors. Other functions such as full-wave rectifiers can also be achieved by switches that are switched on or off according to the phase of the input signal.

There are 6864 electronic switches on the chip that EasyAnalog software is able to program .

There are twenty usable analog zones on the chip. Each zone contains an opamp and five capacitors. Each capacitor is actually made up of a bank of 255 tiny capacitors that can be switched on or off so that each of the five capacitors can be set to 255 different values. The capacitor values are set using "static" switches. There are also "dynamic"

switches on the chip that handle the dynamic switching that makes a capacitor function as if it were a resistor. There are also dynamic switches that switch with respect to the phase of the input signal. In addition, there are lots of static switches that allow the capacitors and opamp within each analog zone to be connected in various ways.

In addition there are switches that allow limited local connections to be made between the components of two zones. (Limited in the sense that there are a limited number of switches and only certain connections are "allowable", i.e., a switch must exist between two nodes for them to be connectable.)

There are also "global wires" on the chip that allow connections between zones that are so far apart that they cannot be connected using the local interconnect scheme.

EasyAnalog software's macro library, "circuits.lib", is composed in part of switch settings that set up the internal connectivity of the opamps and capacitors of one or two zones. EasyAnalog software handles all of the details for you so you don't have to learn about all of the internal switches.

All you have to do is select the *macros (see page 60)* you want to use, connect them with "wires", adjust each macro's parameters using a pop-up menu (right click on a macro and it appears), download the data to the chip and see the results on your oscilloscope.

The Hardware

Connecting the Field-Programmable Analog Array Hardware

First, connect *The POD (see page 11)*

then connect *The BOARD (see page 11)*

Select the correct *serial port (see page 46)*

You should now be ready to design circuits, download their configuration data, and see results on your oscilloscope.

The POD

The download cable connects from the serial port of the computer to the header, JP3, on the Evaluation Board. This cable is used to download from the computer to the MPAA020 the analog array *configuration data (see page 55)* required to implement the circuitry that was designed in EasyAnalog.

The Pod attached to the download cable contains circuitry to convert the RS232-formatted data stream to an SPI-like serial data stream that is compatible with the MPAA020's digital interface.

The Board

The board is provided as part of the FPAA Development Kit. The PCB contains the MPAA020 analog array, supporting circuitry, and various user options for easy circuit prototyping. The following describes these various options and circuitry.

Powering the MPAA 3 Evaluation Board (see page 12)

MPAA020 and Headers (see page 12)

Oscillator and User-defined Oscillator Option (see page 13)

Analog Array Programming Modes and Op Amp Disable (see page 13)

The POR (Power On Reset), RESET, BFR (Boot From ROM), and PWRUP (Power-Up) Push Buttons (see page 13)

PWR ON (Power On), ERR (Error), and END (End of Configuration) LED's (see page 14)

Serial EPROM (see page 14)

The Sallen Key Input Filter and AC IN (for input signals) (see page 14)

Powering the MPAA 3 Evaluation Board

The Evaluation Board may be powered one of three ways.

Note: To utilize the on-board 5 V regulator in 1) and 2) below, a jumper must be installed on the pins of header JP9.

1) Power the board using Banana Plug Connectors, J4 ("PWR") and J5 ("GND"). At least 8 V should be applied to power the on-board 5 V regulator.

2) Power the board using the DC power supply mini connector, J4. The AC to DC power supply should have an output of 9 V DC and a current sourcing capability of 500 mA. An example of such a supply can be purchased through Newark (800-4NEWARK), Stancor part number STA-4190A.

3) By REMOVING the jumper between the pins of header JP9, a 5 V supply may be connected to the left pin of header JP9, labeled "+5V", to directly power the on-board circuitry. As with any analog circuit design, the 5 V supply should be stable and clean (noise-free) for best performance.

MPAA020 and Headers

Each pin of the MPAA020 is connected to a header pin for signal input/monitoring and evaluation purposes. Various input signal instrumentation that can be connected to the headers may include, but is not limited to arbitrary waveform generators, signal generators, dc supplies, sensor/ transducer output signals, etc. The various output signal instrumentation that can be connected to the headers may include, but is not limited to oscilloscopes, multimeters, network and frequency analyzers, etc. Other voltage-controlled circuitry can be connected to the analog array; however, one must remember that the analog array has limited current sourcing/sinking capability (see MPAA020 data sheet for maximum current sinking and sourcing specifications). In such applications, additional interface circuitry should be considered.

The header pins are labeled to indicate the pins of the MPAA020. JP2 is connected to the digital interface pins of the MPAA020 including the pins that are required to download the configuration data stream for the circuit designed using the EasyAnalog Design Software. These pins for the serial data communications are connected to JP3. JP3 is the header to which the Serial Port Download Cable should be attached. The pins of headers JP6, JP7, and JP12 provide probe/ connection points for the analog array's I/O's.

Note: JP2, as labeled, contains a column of +5V (outer column) and Ground (inner column) connections with the digital signal lines connected to the middle column of the header. JP6, JP7, and JP12 contain an inner column/row of Ground-connected pins with the analog array's I/O's being connected to the outer column/row of pins. This configuration is useful for connecting low-inductance probes to the I/O pins of the MPAA020.

Oscillator and User-defined Oscillator Option

The evaluation board contains an on-board 1 MHz oscillator to drive the analog array's internal clock when a jumper is installed between the middle pin and the pin labeled "1 MHz CLK" of header JP8. However, user-defined clock frequencies ranging from 1 kHz to 1 MHz may be implemented by installing a jumper between the middle pin and the pin labeled "EXT CLK". of header JP8. The external clock is input through the BNC connector, J1, which is also labeled "EXT CLK".

Analog Array Programming Modes and Op Amp Disable

The MPAA020 contains four bits in its digital interface to select various programming modes. The modes are selected via switch, SW4. The following table describes the modes that may be implemented on the evaluation board:

From the above table, bit M1 should always be high and bit M0 should always be low to configure the analog array. Bit M2 should always be high when configuring the analog array from either the EasyAnalog Design Software and the Download Cable (a jumper must be installed to connect the middle pin and the pin labeled "CABLE" of JP1) or a serial EPROM (a jumper must be installed to connect the middle pin and the pin labeled "ROM" of JP1). Bit M3 is unused and should be set low.

Mode Bits		Description
[M1]	[M0]	Bits M1 and M0 of Switch SW4
1	0	BFR Mode - Boot From ROM, serial data, low pin count EPROM generates own addresses
Mode Bits		Description
[M3]	[M2]	Bits M3 and M2 of Switch SW4
0	1	Use external clock for configuration
0	0	Use MPAA internal clock for configuration (not required for evaluation board use)

In addition to the configuration modes, switch SW4 contains an additional bit, "OP DIS", to disable the op amps and op amp buffers of the analog array. Switch "OP DIS" high to disable the op amps. Upon switching "OP DIS" low, the op amps are enabled.

The POR (Power On Reset), RESET, BFR (Boot From ROM), and PWRUP (Power-Up) Push Buttons

These buttons have dedicated functionalities as described below:

RESET, SW1

Pushing the RESET button, SW1, clears the MPAA020 configuration memory and begins a configuration sequence, as determined by the programming mode bits when the RESET button is released.

BFR (Boot From ROM), SW2

Pushing the BFR button, SW2, clears the MPAA020 configuration memory and begins a configuration sequence, as determined by the programming mode bits.

PWRUP (Power-Up), SW3

By holding down the PWRUP button during a configuration sequence, user outputs and inputs are disabled until the button is released. This button has the same effect as "OP DIS" (Op amp disable) of SW4.

PWR ON (Power On), ERR (Error), and END (End of Configuration) LED's

- The PWR ON LED is illuminated when power is supplied to the board.

- The ERR LED is illuminated when, during a configuration sequence, a serial data transmission checksum error has occurred. If a serial data transmission checksum error occurs, download the configuration file again (from the EasyAnalog Design Software and the download pod). If using a serial EPROM and a checksum error occurs, cycle power or press one of the POR, RESET, or BFR buttons to re-load the configuration code from the serial EPROM.

- The END LED turns off upon successful configuration of the analog array. The analog circuit design created using the EasyAnalog Design Software is now fully functional in the MPAA020.

Serial EPROM

An optional serial EPROM, inserted into socket U1, is used to load an analog circuit configuration into the MPAA020 upon power-up (i.e. without the necessity of using the EasyAnalog Design Software and the Download Cable). Make sure when configuring the MPAA020 from a serial EPROM that a jumper is installed from the middle pin to the pin labeled "ROM" of header JP1.
A recommended device is Motorola's 8-pin MPA1765 (64K) serial EPROM. The DIP version, MPA1765P, is required for use with the evaluation board.

The Sallen Key Input Filter and AC IN (for input signals)

To demonstrate optional external anti-aliasing filtering of an input signal to the analog array, one of the analog array's input buffer amplifiers (pins LDX, LDY, and LDZ) is dedicated as an input (using the EasyAnalog Design Software, user must turn on the input buffer and ensure the switch between LDX and LDY is OPEN) from the output of the 4th order 200kHz (3dB point) anti-aliasing Sallen Key input filter. The output of the filter is located at the LDZ pin.

A BNC connector, J7, is provided for the input signal to the anti-aliasing filter. The input signal is referenced to earth ground (GND) as required by some signal generators. All input signals connected to J7 are AC coupled.

If AC coupling of the input signal is not desired, the input signal may be input directly to the anti-aliasing filter by attaching the signal to the top pin of header JP11 (JP11 is also labeled "SIGIN"). The ground connection of the input signal may then be attached to either signal ground (bottom pin of header JP11) or to earth ground. The intention of JP11 is to allow the input signals that are referenced to signal ground to be measured using low inductance probes.

The Sallen Key Output Filter and Output Coupling Options

To demonstrate optional external filtering of the analog array's output, one of the analog array's input buffer amplifiers (pins RDX, RDY, and RDZ) is dedicated as an output (using the EasyAnalog Design Software, user must turn on the input buffer and ensure the switch between RDX and RDY is OPEN) from the 4th order 200kHz (3dB point) Sallen Key output filter.

The output of the Sallen Key filter can be AC or DC coupled, as may be desired or required of some instrumentation.

To measure the DC-coupled signal output from the Sallen Key filter, remove all jumpers from headers JP4 and JP5. Measure the DC-coupled signal on the pin of header JP5 labeled "DCOUT".

To measure the AC-coupled signal output from the Sallen Key filter, install a jumper from the pin labeled "DCOUT" of JP5 to the pin directly above it of JP4 (the left pin of header JP4). Measure the AC-coupled signal on the pin of header JP4 labeled "ACOUT".

A ground pin (the right pin of header JP5) is available for measuring either the DC or AC coupled signal using low inductance probes.

Stereo Mini Plugs

Perhaps the most specialized feature of the evaluation board is the stereo mini plugs, J9 and J10. These may be used for any stereo (or mono) input signal (e.g. the output of a stereo walkman, CD player, etc.) that requires additional signal conditioning, filtering, etc. For maximum flexibility, J9 and J10 are not hardwired to specific pins of the MPAA020; rather, via headers JP14 and JP15, the signals from J9 and J10 may be connected to any of the pins on headers JP6, JP7, and JP12 via the square-pin-to-square-pin jumper cables supplied with the evaluation board. For each of the headers, JP14 and JP15, the right channel input signal is connected to the leftmost pin, ground is connected to the middle pin, and the left channel input signal is connected to the rightmost pin.

Signal Ground (SGND) Connection Options

There are three options to generating the signal ground for the analog array's operation:

- Normally, the signal ground for the MPAA020 is 2.5 V as generated internally to the MPAA020 and measurable at the pin labeled "VMR" on header JP7. To use the internally generated signal ground reference of 2.5 V, DO NOT install a jumper across header JP11.
- Since the internally generated reference, VMR, does not have high current sourcing capability for applications that require additional instrumentation/circuitry to use the signal ground reference, the evaluation board also contains an on-board 2.5 V reference, U2, for such applications. To use the 2.5 V reference for signal ground, install a jumper across JP10, AND, via the EasyAnalog Design Software, disable the internally generated reference, VMR.
- For the third option, a user-defined signal ground may be implemented using an external power supply, voltage reference, etc., via the Banana Plug Connector, J6 (also called "SGND"). To implement a user-defined signal ground, connect the desired voltage (between 2.25V and 2.75V) to the SGND Banana Plug Connector, and most importantly, DO NOT install a jumper across header JP11

AND, via the EasyAnalog software, disable the internally generated reference, VMR.

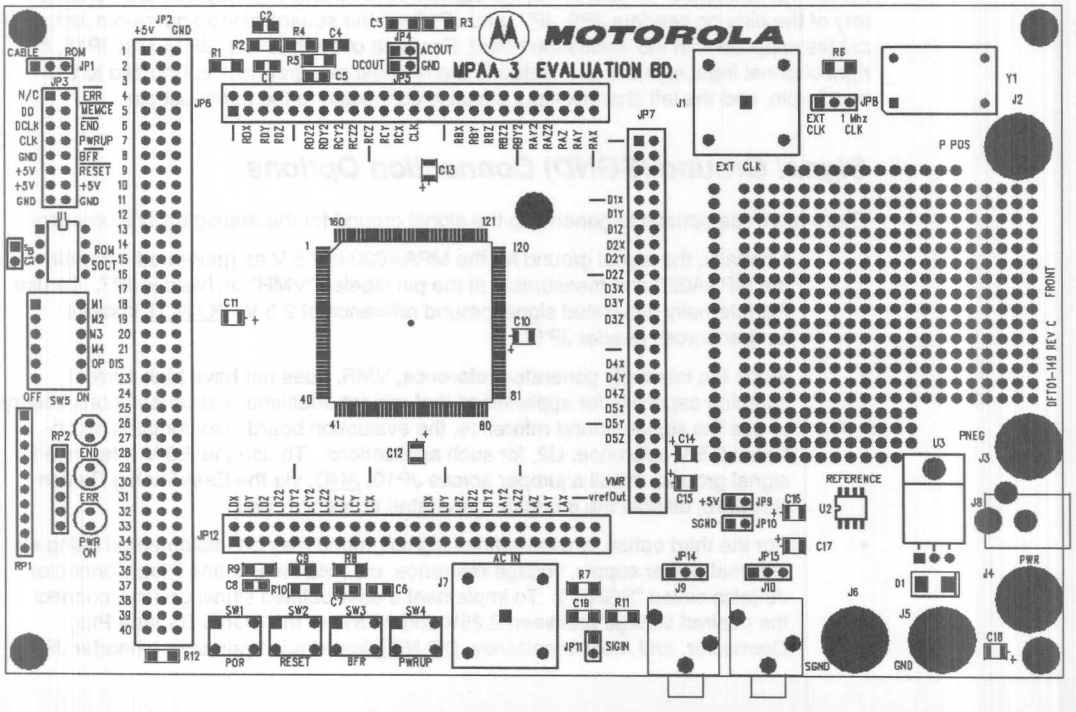
In all three options above, whether the signal ground reference is created via the VMR, the on-board voltage reference, or externally, the Banana Plug Connector, SGND, is available to monitor/connect the signal ground reference to external instrumentation/circuitry.

P POS, P NEG and the Wire-Wrap Area

The Banana Plug Connectors, P POS and P NEG, are connected to the top-most row and bottom-most row, respectively, of the wire wrap area. These P POS and P NEG "power supply buses" are independent of the evaluation board power and ground connections so that any required voltage for circuitry built on the wire-wrap area can be implemented (i.e. split supplies, higher voltage supplies, etc.).

The wire-wrap area can be used to implement any circuitry in addition to the circuitry already contained in the evaluation board. The single column header directly left of the wire-wrap area may be used to connect any signals from circuitry on the wire-wrap area to any of the pins on headers JP6, JP7, and JP12 via the square-pin-to-square-pin jumper cables supplied with the evaluation board. For ease of use, each row of the single column header is electrically connected to the corresponding row of the last, leftmost column of the wire wrap area (see Board Schematic).

Board Image



EasyAnalog Overview

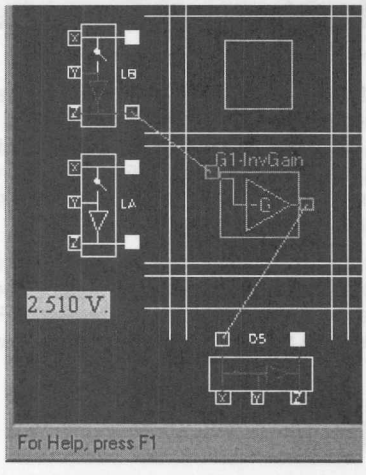
Lets do a circuit

The quickest way to put a circuit up on the screen is to use one of the example circuits.

First, if you haven't already done so, you should *connect the hardware (see page 11).*

Lets try a simple *inverting gain stage (see page 61).*

1. In the EasyAnalog software window, use the left mouse button and click on the "file" menu item. This brings up the "file" menu list.
2. Click on the "Open . . ." menu item in the "file" menu list. This brings up a file selection window.
3. You will see an "Examples" folder in the window. Use the mouse to place the cursor over the "Examples" folder and double-click the left mouse button (i.e., press the left mouse button quickly twice in succession.) This will display the list of example circuits in the window.
4. Locate the "Inverting Gain Stage.ckt" entry, place the cursor over it, and double-click the left mouse button. The gain stage circuit will be loaded into EasyAnalog software and displayed in the lower left corner of the EasyAnalog software display screen as shown below.



Inverting gain stage example as displayed in EasyAnalog software

Note that the green object labeled "Gain-Inv" is placed over one of the twenty chip zones (squares on the screen) indicating that the zone is being used. Also note that the Gain-

Inv macro receives its input from the input terminal of the IO Cell labeled LB and sends its output to the output terminal of the IO Cell labeled D5.

In the LB IO Cell, the opamp has been turned on so that you can connect your input to one of the terminals labeled LBY or LBZ on the Motorola FPAA Evaluation Board.

If you use the LBY terminal for your input, it will be buffered by the opamp so that the FPAA does not load down the circuit that is generating your input signal.

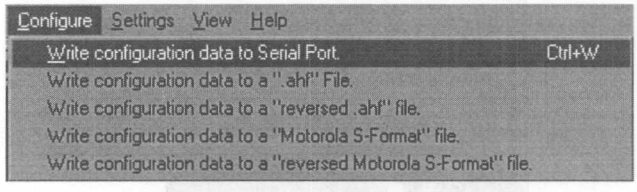
The input should consist of a sine wave (or other periodic signal). The voltage of the input signal should not exceed 5 volts peak-to-peak of course (the chip's signal ground is at +2.5 volts). A signal generator can be used to provide this input.

In the D5 IO Cell, the switch and the opamp have been turned on so that the user can inspect the output from the full-wave rectifier by attaching an oscilloscope probe to one of the terminals labeled D5X, D5Y, or D5Z on the Motorola FPAA Evaluation Board.

If you use D5Z as the output terminal it will buffer the FPAA circuit, i.e., it will prevent the external load from loading down the FPAA output circuit.

The default gain setting for the Gain-Inv macro is "-1", i.e., the gain is "1" and the output signal will be inverted.

Assuming you have your hardware all set up, and assuming that you have selected the correct *Serial Port (see page 46)*, and assuming that the signal generator and oscilloscope are attached to the proper terminals, we are now ready to download the *configuration data (see page 55)* to the chip. We do this by selecting the menu item "Configure/Write configuration data to Serial Port" as shown below.



Download data to the Serial Port.

There will be a bell signifying start of download. When you hear a second bell, download is complete and the circuit should immediately begin working as an inverting gain stage, i.e., if you feed in a sine wave to terminal LBY from the signal generator, you should see an inverted sine wave on the output terminal, D5Z. The download takes approximately 3 seconds. If you don't hear the second bell you will want to investigate possible *serial port communications problems (see page 47)*.

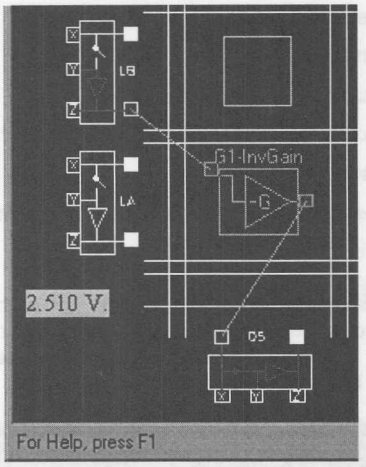
The gain of the Gain-Inv macro defaults to "-1". The "*Gain stage Set Macro Parameters (see page 52)*" dialog box can be used to adjust the gain of the Gain-Inv macro. You can adjust the gain then again download the data to the Serial Port to immediately see the effect of the new gain value on the amplitude of the output wave-form.

Lets try a full-wave rectifier.

Here is how to load the full-wave rectifier example circuit:

1. In the EasyAnalog software window, use the left mouse button and click on the "file" menu item. This brings up the "file" menu list.

2. Click on the "Open..." menu item in the "file" menu list. This brings up a file selection window.
3. You will see an "Examples" folder in the window. Use the mouse to place the cursor over the "Examples" folder and double-click the left mouse button (i.e., press the mouse key quickly twice in succession.) This will display the list of example circuits in the window.
4. Locate the "Full-wave-rectifier.ckt" entry, place the cursor over it and double-click the left mouse button. The full-wave rectifier circuit will be loaded into EasyAnalog software and displayed in the lower left corner of the EasyAnalog software display screen as shown below.



Full-Wave Rectifier example as displayed in EasyAnalog software

Note that the green object labeled "Rectifier" is placed over one of the twenty chip zones indicating that the zone is being used. Also note that the rectifier macro receives its input from the IO Cell labeled LB and sends its output to the IO Cell labeled D5.

In the LB IO Cell, the opamp has been turned on so that the user can connect his input to one of the terminals labeled LBY or LBZ on the Motorola FPAA Evaluation Board. This input should consist of a sine wave (or other periodic signal that is to be rectified). The voltage of the input signal should not be outside of the range +/- 2.5 Volts of course. (Signal ground is at +2.5 Volts.) A signal generator can be used to provide this input. If you use LBY for your input, it will be buffered by the opamp so that the FPAA does not load down the circuit that is generating your input signal.

In the D5 IO Cell, the switch and the opamp have been turned on so that the user can inspect the output from the full-wave rectifier by attaching an oscilloscope probe to one of the terminals labeled D5X, D5Y, or D5Z on the Motorola FPAA Evaluation Board. If you use D5Z as the output terminal it will buffer the FPAA circuit, i.e., it will prevent the external load from loading down the FPAA output circuit.

If you put the cursor over the green object labeled "Rectifier" and then press the RIGHT mouse button, the "*Full Wave Rectifier Set Macro Parameters (see page 49)*" popup dialog box will appear. You can use it to adjust the filter corner frequency and pass band gain of the rectifier macro.

You may now download your configuration file to the Field-Programmable Analog Array as you did in the *first example (see page 19)*, and you should see a rectified version of your input signal on the oscilloscope trace that is displaying the output signal.

Brief Tutorial - Creating a New Circuit

At any point, you can choose *File/New (see page 33)* to clear all contents of the main window.

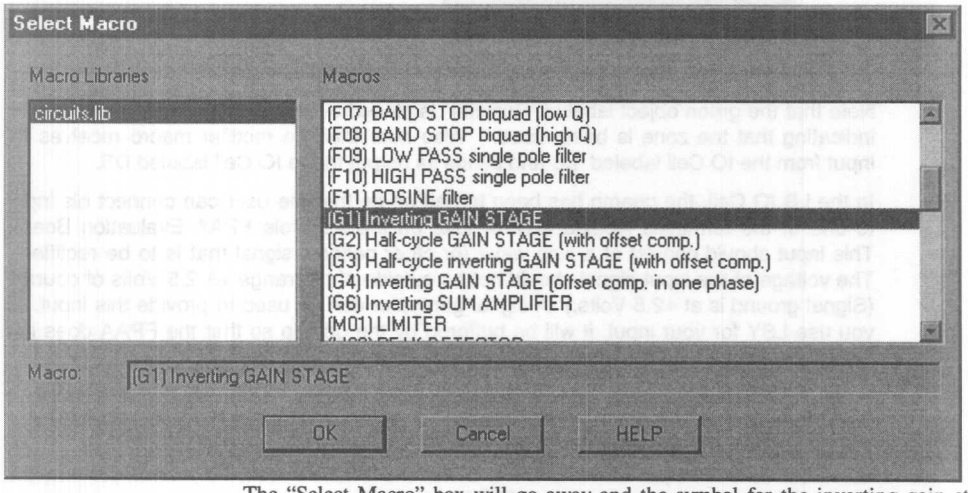
To create a new circuit, you will

- Select and place macros,
- Set macro parameters,
- Connect the macros and IO Cells with wires,
- Download the configuration file to the Field-Programmable Analog Array,
- Observe the results.

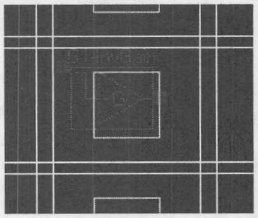
Lets create the circuit for the *first example (see page 19)* from scratch:

First we select the macro:

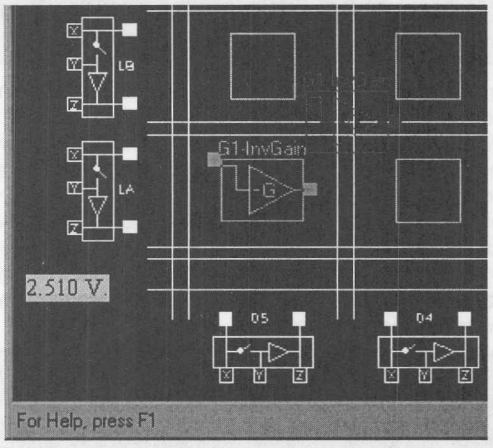
- Click on the  symbol in the tool bar just above the main window to bring up the Select Macro window. Use the scroll bar on the right to scroll down to where the Gain Stage macro may be seen. Then place the mouse pointer over it and do a left mouse button double-click:



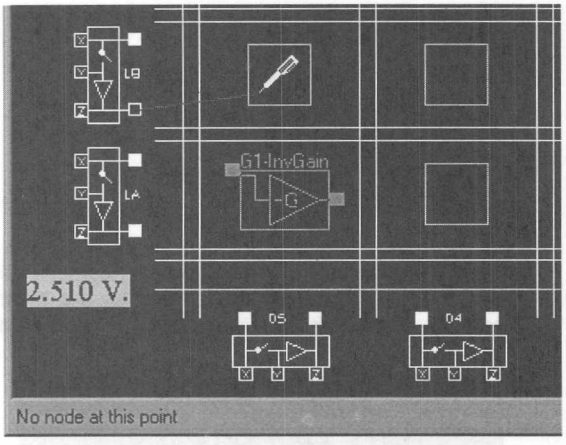
- The "Select Macro" box will go away and the symbol for the inverting gain stage will be displayed on the screen:



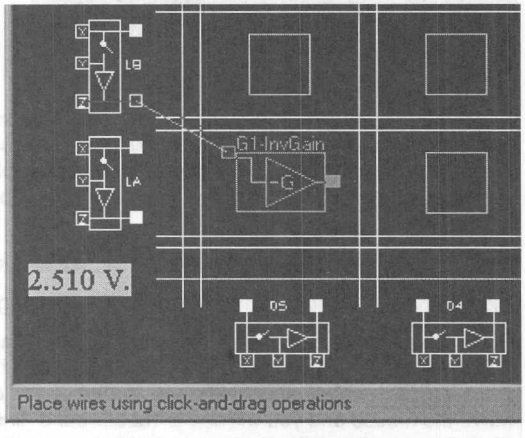
- As you move the mouse, the gain stage symbol will move. Place it over the *zone marker (see page 27)* in the lower left corner of the main screen. (You could place it on any of the 20 zones but, for this tutorial, we choose the lower left corner.) Then click the left mouse button. The image will change to green and will have thus been "placed":



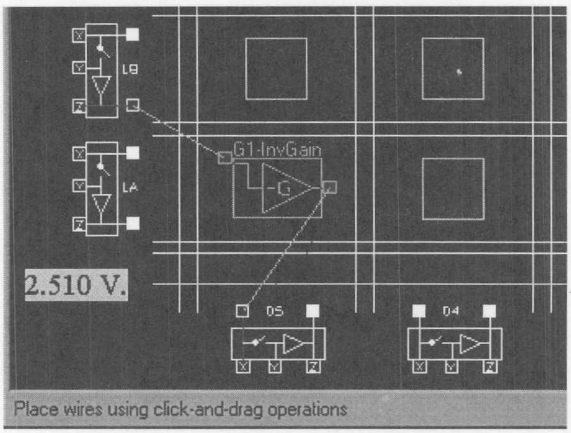
- Notice that there is now a green image and a red image being displayed. The red image can be moved about by moving the mouse and can be used to place another "instance" of the gain stage. For this example, we will not place another instance but instead we will enter "*wire mode (see page 41)*" to begin wiring the circuit.
- Enter "wire mode" either by typing a "w" on the keyboard or by left mouse-clicking the  symbol. The cursor image will change to a drawing pen symbol.
- Place the tip of the cursor over the "output" (lower) terminal of the IO Cell labeled "LB". Press the left mouse button down and hold it down. The interior of the white output terminal should change to blue if the cursor was placed properly. Move the mouse and you will see a "rubber-band" line from the terminal to the tip of the cursor that follows the cursor where ever it moves. As you move the cursor over the vertical and horizontal global wires and other terminals on the screen, note the highlighting of global wires and terminals that occurs. Also note the messages that appear in the lower left corner of the screen, i.e., in the "status bar".



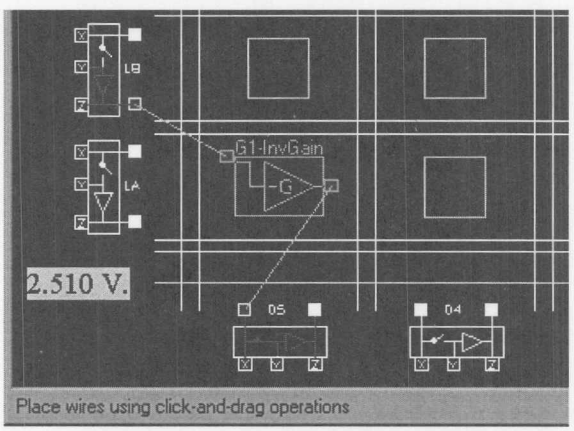
- Circuit node connectivity in the Field-Programmable Analog array is accomplished by setting switches within the chip. Some nodes cannot be directly connected because no switch exists that would directly connect them. The messages tell you continuously whether you can or cannot connect to the site the cursor is over. It also provides the reason why you cannot connect to some particular site. Place the cursor directly over the solid green square (contact) on the left side of the gain stage. The interior of the green square will turn blue. Release the mouse button. A blue wire will appear in place of the red rubber band line and the interiors of both contacts will turn red indicating that a connection has been made.



- Now connect the output terminal of the gain stage to the input terminal of the IO Cell labeled "D5":



- Now turn on the appropriate opamps and switches by placing the cursor directly over them and clicking the left mouse button:

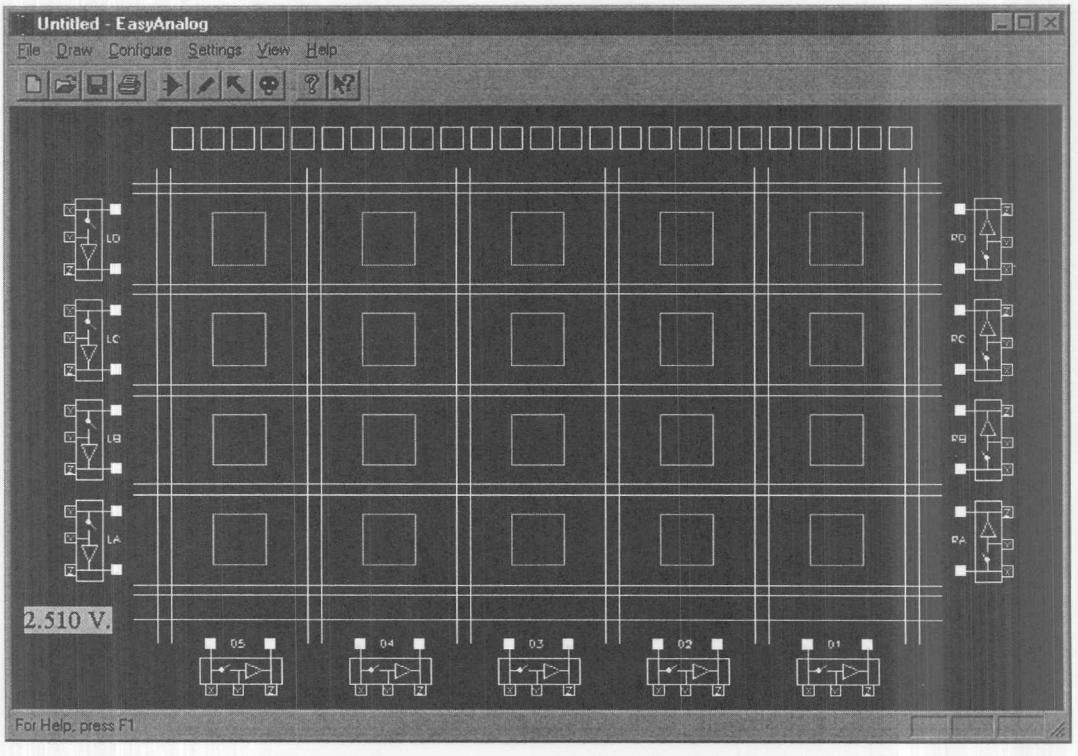


- That completes construction of the circuit. To see this circuit in operation, we would input a signal such as a sine wave to the "Y" pin of the "LB" IO Cell and attach an oscilloscope probe to the "Z" pin of the "D5" IO Cell, *download the configuration data (see page 57)*, then look at the oscilloscope screen. The default gain of the inverting opamp is "-1" (minus because its inverting). We could **right** mouse click on the center of the green gain stage symbol to bring up the "*Set Macro Parameters (see page 48)*" dialog window to alter the value of the gain. We would close the dialog window by clicking on "OK" to store the new gain value then again download the configuration data, and look at the oscilloscope to see the effect of the new gain value.

Software Reference

EasyAnalog Screen

Main Screen



Zone Markers

A Zone Marker is simply one of twenty rectangles that are drawn on the *main screen (see page 27)* that shows where one of the primary chip cells is located. When you *select a macro (see page 38)* and place it, you place it on top of one (or more if it is a multi-cell macro) of the zone markers.

Global Horizontal Wires

Global horizontal wires are part of the global interconnect scheme and can be used to connect components (macros, IO Cells, etc.) that are too far apart to be reached via local interconnect. (Remember, there are a limited number of switches on the chip so direct connections between any two particular elements are not always possible.) Use of global wiring will allow you to connect elements that could not otherwise be reached.

Horizontal and vertical global wires may be connected in "*wire mode (see page 41)*" by a left mouse down-click on one wire, hold down and drag, and a left mouse up-click on the other wire. Alternatively, a left mouse down then up click on a horizontal/vertical global wire junction will connect the junction.

To disconnect global wire junctions, get into "*delete mode (see page 32)*" and left mouse-click (down, then up) the red dot at the junction. See *Main Screen (see page 27)* for a view of global and vertical wires.

Global Vertical Wires

Global vertical wires are part of the global interconnect scheme and can be used to connect components (macros, IO Cells, etc.) that are too far apart to be reached via local interconnect. (Remember, there are a limited number of switches on the chip so direct connections between any two particular elements are not always possible.) Use of global wiring will allow you to connect elements that could not otherwise be reached.

Horizontal and vertical global wires may be connected in "*wire mode (see page 41)*" by a left mouse down-click on one wire, hold down and drag, and a left mouse up-click on the other wire. Alternatively, a left mouse down then up click on a horizontal/vertical global wire junction will connect the junction.

To disconnect global wire junctions, get into "*delete mode (see page 32)*" and left mouse-click (down, then up) the red dot at the junction. See *Main Screen (see page 27)* for a view of global and vertical wires.

Voltage Reference (VREF)

The chip contains a voltage reference wire that can be set to one of 256 voltage values. The wire behaves as a horizontal global wire. The voltage can be set using the *Set Electrical Parameters (see page 53)* dialog window. Note that the value displayed in the parameters window is not a measured value, it is an estimated value. You should measure the value and set it according to your measurements if you wish to be precise. See *Main Screen (see page 27)* for a view of the VREF horizontal wire.

File Name

File names in EasyAnalog software follow the Windows 95 conventions. The suffix assigned to circuit files is ".ckt".

Other files are produced by EasyAnalog software when you choose to write the configuration data to a file. Configuration data files will be written into the same directory where your ".ckt" file exists, i.e., your working directory. (The working directory for a particular ".ckt" file is established when you save or open a file.) Four types of configuration files are possible that may be used to program an EPROM with the circuit you have designed. Four types of files were provided because some of the third-party

EPROM programmer software packages handle one but not the other format and because serial EPROMs and parallel EPROMs have different internal bit order requirements. Assume you have a circuit that you named "gain.ckt". The four types of files would be:

"gain.ahf", an *"AHF or ASCII Hex Format" file* *(see page 56)* suitable for programming a parallel EPROM

"gain-r.ahf", an *"Reversed AHF or ASCII Hex Format"* *(see page 55)* file suitable for programming a serial EPROM

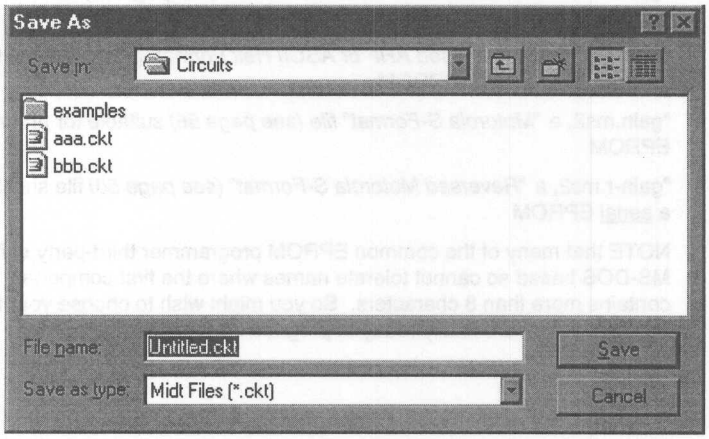
"gain.ms2, a *"Motorola S-Format" file* *(see page 56)* suitable for programming a parallel EPROM

"gain-r.ms2, a *"Reversed Motorola S-Format"* *(see page 56)* file suitable for programming a serial EPROM

NOTE that many of the common EPROM programmer third-party software packages are MS-DOS based so cannot tolerate names where the first component of the name contains more than 8 characters. So you might wish to choose your file name with this in mind when you are finally ready to program an EPROM.

Dialog Boxes

Save As Dialog Box



File Name

Type a new filename to save a circuit with a different name. EasyAnalog software will always assign the extension ".ckt" to the filename you specify.

Save In

Select the directory in which you want to store the circuit.

Save as type

EasyAnalog software will assign type ".ckt" to your file.

Print Dialog Box

Print Progress Dialog

The Printing dialog box is shown during the time that EasyAnalog software is sending output to the printer. The page number indicates the progress of the printing.

To abort printing, choose Cancel.

Commands

Buffers/Switches Command

This allows you to left-click on the buffer opamps and switches in the IO Cells around the periphery of the chip and toggle them on or off. It also allows all of the *Wire mode (see page 41)* operations.

In fact, this mode is identical to wire mode except for the instructions that appear in the lower left corner of the display. This command was added primarily for user clarification regarding how to toggle the buffer opamps and switches, i.e., we wanted you to be able to figure it out without reading this manual if possible.

Chip Settings Command

The *Set Electrical Parameters (see page 53)* dialog box allows you to set many of the chip's operational parameters. These parameters are:

1. The frequency of the master clock you are using.
2. The frequencies of four divide-down clocks derived from the master clock frequency, clock0, clock1, clock2, and clock3. (Each macro that you have placed may be assigned to use one of these four clocks in the "*Set Macro Parameters (see page 48)*" dialog window for that macro.

3. The chip's IBIAS value, a number from 0 to 15, that sets the bias current for the 20 primary opamps. (Nominal value should be 8).
4. *Voltage Reference (see page 28).* A voltage source that can be adjusted and used.
5. Enable *VREF (see page 28).* Turns the VREF voltage source on or off.
6. Enable VMR. Turns the internal Mid-Range Voltage source on or off. This is essentially signal ground. You should disable VMR if you wish to use an external signal ground. (There is an chip pin labeled VMR and a terminal on the board labeled VMR.)

Configure menu commands

The Configure menu offers the following commands:

Command	Function
Write configuration data to Serial Port (see page 57)	**Write the current circuit's configuration data to the POD (which in turn writes the data to the FPAA chip on the board) via the serial port selected in the Settings/Serial Port menu.**
Write configuration data to a ".ahf" file (see page 56)	**Write the current circuit's configuration data to a file with the same name as the circuit name but with a ".ahf" suffix.**
Write configuration data to a "-r.ahf" file (see page 55)	**Write the current circuit's configuration data to a file with the same name as the circuit name but with a "-r.ahf" suffix.**
Write configuration data to a ".ms2" file (see page 56)	**Write the current circuit's configuration data to a file with the same name as the circuit name but with a ".ms2" suffix.**
Write configuration data to a "-r.ms2 " file (see page 56)	**Write the current circuit's configuration data to a file with the same name as the circuit name but with a "-r.ms2" suffix.**

Delete Command

The "delete" mode allows you to delete wires, global wire junction connections and macros. It may also be used to toggle the IO Cell buffer opamps and switches.

To delete a wire, place the cursor anywhere over the wire and click the left mouse button. The wire is deleted on the up-click. To delete a global wire junction connection, place the cursor over the junction and click the left mouse button. The junction is deleted on the up-click. To delete a macro, place the cursor over the macro and click the left mouse button. The macro is deleted and so are any wires that were connected to the macro.

You can exit "delete" mode by pressing the escape key (Esc) or by selecting another mode from the "Draw" menu or by pressing one of the following tool-bar buttons:

Shortcuts

Toolbar: 

Keys: D

Draw menu commands

The Draw menu offers the following commands:

Command	Function
Macro (see page 38)	**Select a macro from a list of macros.**
Wires (see page 41)	**Enable point-to-point click-and-drag wiring.**
Buffers/Switches (see page 31)	**Allow IO Cell opamps and switches to be toggled.**
Shift Macro (see page 42)	**Allow placed macros to be moved to other locations.**
Delete (see page 32)	**Allow deletion of macros and wires.**

File menu commands

The File menu offers the following commands:

Command	Function
New (see page 33)	**reates a new circuit.**
Open (see page 34)	**pens an existing circuit.**
Save (see page 37)	**aves an opened circuit using the same file name.**
Save As (see page 37)	**aves an opened circuit to a specified file name.**
Print (see page 34)	**rints a circuit.**
Print Preview (see page 35)	**isplays the circuit on the screen as it would appear printed.**
Print Setup (see page 36)	**elects a printer and printer connection.**
Exit (see page 37)	**Exits EasyAnalog software.**

New command (File menu)

Use this command to create a new circuit in EasyAnalog software.

You can open an existing circuit with the *Open command (see page 34)*

Shortcuts

Toolbar: 

Keys: CTRL+N

Open command (File menu)

Use this command to open an existing circuit in a new window.

See the "*File Open Dialog Box (see page 34)*".

See *Window 1, 2, ... command (see page 37)*.

You can create new circuits with the *New command (see page 33)*

Shortcuts

Toolbar: 

Keys: CTRL+O

File Open dialog box

The following options allow you to specify which file to open:

File Name

Type or select the filename you want to open. This box lists files with the extension you select in the List Files of Type box.

List Files of Type

Select the type of file you want to open:

The only allowable file type that can be loaded into EasyAnalog software is a "circuit" file having a suffix of ".ckt".

Drives

Select the drive in which EasyAnalog software stores the file that you want to open.

Directories

Select the directory in which EasyAnalog software stores the file that you want to open.

Network...

Choose this button to connect to a network location, assigning it a new drive letter.

Print command (File menu)

Use this command to print a document. This command presents a *Print dialog box (see page 35)*, where you may specify the range of pages to be printed, the number of copies, the destination printer, and other printer setup options.

Shortcuts

Toolbar: 

Keys: CTRL+P

Print dialog box

The following options allow you to specify how the document should be printed:

Printer

This is the active printer and printer connection. Choose the Setup option to change the printer and printer connection.

Setup

Displays a *Print Setup dialog box (see page 36)*, so you can select a printer and printer connection.

Print Range

Specify the pages you want to print:

All	**Prints the entire document.**
Selection	**Prints the currently selected text.**
Pages	**Prints the range of pages you specify in the From and To boxes.**

Copies

Specify the number of copies you want to print for the above page range.

Collate Copies

Prints copies in page number order, instead of separated multiple copies of each page.

Print Quality

Select the quality of the printing. Generally, lower quality printing takes less time to produce.

Print Preview command (File menu)

Use this command to display the active document as it would appear when printed. When you choose this command, the main window will be replaced with a print preview window in which one or two pages will be displayed in their printed format. The *print preview toolbar (see page 35)* offers you options to view either one or two pages at a time; move back and forth through the document; zoom in and out of pages; and initiate a print job.

Print Preview toolbar

The print preview toolbar offers you the following options:

Print

Bring up the print dialog box, to start a print job.

Next Page

Preview the next printed page.

Prev Page

Preview the previous printed page.

One Page / Two Page

Preview one or two printed pages at a time.

Zoom In

Take a closer look at the printed page.

Zoom Out

Take a larger look at the printed page.

Close

Return from print preview to the editing window.

Print Setup command (File menu)

Use this command to select a printer and a printer connection. This command presents a *Print Setup dialog box (see page 36)*, where you specify the printer and its connection.

Print Setup dialog box

The following options allow you to select the destination printer and its connection.

Printer

Select the printer you want to use. Choose the Default Printer; or choose the Specific Printer option and select one of the current installed printers shown in the box. You install printers and configure ports using the Windows Control Panel.

Orientation

Choose Portrait or Landscape.

Paper Size

Select the size of paper that the document is to be printed on.

Paper Source

Some printers offer multiple trays for different paper sources. Specify the tray here.

Options

Displays a dialog box where you can make additional choices about printing, specific to the type of printer you have selected.

Network...

Choose this button to connect to a network location, assigning it a new drive letter.

Save As command (File menu)

Use this command to save and name the active circuit. EasyAnalog software displays the *Save As dialog box (see page 30)* so you can choose a *file name (see page 28)* your circuit.

To save a circuit with its existing name and directory, use the *Save command (see page 37)*

Save command (File menu)

Use this command to save the active circuit to its current name and directory. When you save a circuit for the first time, EasyAnalog software displays the *Save As dialog box (see page 30)* so you can name your circuit. If you want to change the name and directory of an existing circuit before you save it, choose the *Save As command (see page 37)*

Shortcuts

Toolbar: 

Keys: CTRL+S

1, 2, ... command (Window menu)

EasyAnalog software displays a list of the most recent previously opened circuits at the bottom of the Window menu. Choose a circuit to cause it to be loaded and displayed.

Exit command (File menu)

Use this command to end your EasyAnalog software session.

EasyAnalog software prompts you to save circuits with unsaved changes.

Shortcuts

Mouse: Double-click the application's Control menu button.

Keys: ALT+F4

Help menu commands

The Help menu offers the following commands, which provide you assistance with this application:

Command	Function
Help Topics (see page 38)	**ffers you an index to topics on which you can get help.**
About EasyAnalog software (see page 38)	**isplays the version number of this application.**

Help Using Help Command

Context Help command

Use the Context Help command to obtain help on some portion of EasyAnalog software. When you choose the Toolbar's Context Help button, the mouse pointer will change to an arrow and question mark. Then click somewhere in the EasyAnalog software window, such as another Toolbar button. The Help topic will be shown for the item you clicked.

Shortcut

Keys: SHIFT+F1

Index command (Help menu)

Use this command to display the opening screen of Help. From the opening screen, you can jump to step-by-step instructions for using EasyAnalog software and various types of reference information.

Once you open Help, you can click the Contents button whenever you want to return to the opening screen.

About command (Help menu)

Use this command to display the copyright notice and version number of your copy of EasyAnalog software.

Macro Command

Displays the *select macro dialog box (see page 59)* so that you can choose a macro to place.

Shortcuts

Toolbar: 

Keys: M

Menu Commands

File menu (see page 33)

Draw menu (see page 33)

Configure menu (see page 32)

Settings menu (see page 42)

View menu (see page 39)

Help menu (see page 37)

Preferences Command

Allows you to set preferences using the *Set Preferences Dialog Window (see page 54)*.

Refresh Command

The Refresh command redraws causes the EasyAnalog software window to be redrawn.

Shortcuts

Keys: R

Serial Port Command

Allows you to select the proper serial port for downloading chip configuration data to the POD. Your PC should have a serial port. Different PC's may have different active serial ports. EasyAnalog software will use the port you select using *the Serial Port Selection Dialog Window (see page 46)*.

View menu commands

The View menu offers the following commands:

Command	Function
Toolbar (see page 40)	**hows or hides the toolbar.**
Status Bar (see page 39)	**hows or hides the status bar.**
Zoom in (see page 41)	**Zoom in closer to the drawing in the EasyAnalog software main window.**
Zoom out (see page 42)	**Zoom out further from the drawing in the EX-Analog main window.**
Zoom fit (see page 41)	**Resize the drawing in the EasyAnalog software main window so that it just fits within the main window region of the screen.**
Refresh (see page 39)	**Redraw everything in the main window.**

Status Bar command (View menu)

Use this command to display and hide the Status Bar, which describes the action to be executed by the selected menu item or depressed toolbar button, and keyboard latch state. A check mark appears next to the menu item when the Status Bar is displayed.

See *Status Bar (see page 39)* for help on using the status bar.

Status Bar

The status bar is displayed at the bottom of the EasyAnalog software window. To display or hide the status bar, use the Status Bar command in the View menu.

The left area of the status bar describes actions of menu items as you use the arrow keys to navigate through menus. This area also shows messages that describe the

actions of toolbar buttons as you depress them, before releasing them. If after viewing the description of the toolbar button command you wish not to execute the command, then release the mouse button while the pointer is off the toolbar button.

The Status Bar in EasyAnalog software is also used to communicate the instantaneous status of wiring operations as you move the mouse with the left button held down in *wire mode (see page 41)*.

The Status Bar in EasyAnalog software also contains the path name of *configuration files (see page 32)* immediately after they are written.

The right areas of the status bar indicate which of the following keys are latched down:

Indicator	Description
CAP	The Caps Lock key is latched down.
NUM	The Num Lock key is latched down.
SCRL	The Scroll Lock key is latched down.

Toolbar command (View menu)

Use this command to display and hide the Toolbar, which includes buttons for some of the most common commands in EasyAnalog software, such as File Open. A check mark appears next to the menu item when the Toolbar is displayed.

See *Toolbar (see page 40)* for help on using the toolbar.

Toolbar

The toolbar is displayed across the top of the application window, below the menu bar. The toolbar provides quick mouse access to many tools used in EasyAnalog software,

To hide or display the Toolbar, choose Toolbar from the View menu (ALT, V, T).

<< Add or remove toolbar buttons from the list below according to which ones your application offers. >>

Click	To
	Open a new circuit.
	Open an existing circuit. EasyAnalog software displays the Open dialog box, in which you can locate and open the desired file.
	Save the active circuit or template with its current name. If you have not named the circuit, EasyAnalog software displays the Save As dialog box.
	Print the active circuit.
	Switch to “select and place macro” mode
	Switch to “draw wires” mode
	Switch to “macro shift (move)” mode

 Switch to "delete selected macros and wires" mode

 Bring up this help file

 Get context sensitive help

Wires Command

Switches to the "wire" mode so that you can connect your circuit components with wires. Circuit components can be global wires, IO Cell contacts, macro contacts or the VREF line. Connections are made by placing the cursor over a contact or global wire, pressing the left mouse button and moving the mouse with the button held down. A rubber band line will follow the cursor. When the cursor is over the contact or global wire that is to receive the connection, you should release the left mouse button. If the connection was valid, the connection is made. Note that continuous validity checks are made as you move the rubber band line.

As the rubber band line is moved over the objects on the screen, the instantaneous results of the validity checks are displayed in the lower left corner of the display. The chip has a limited number of allowable connections because it has a limited number of switches. This means that you may not be able to directly connect two particular objects but you should be able to connect them via the global wires.

Note that the junctions of the global wires are treated specially, i.e., by clicking directly on a junction, you can cause the junction to be connected. You cannot use the junction to connect to some other object however. You have to connect other objects to a global wire using a contact point that is at least some small distance away from the junction.

You can exit "wire" mode by pressing the escape key (Esc) or by selecting another mode from the "Draw" menu or by pressing one of the following tool-bar buttons:

Shortcuts

Toolbar: 

Keys: W

Zoom Fit Command

The Zoom Fit command sizes the image to fit the EasyAnalog software main window.

Shortcuts

Keys: F

Zoom In Command

The Zoom In command magnifies the image displayed in the EasyAnalog software main window.

Shortcuts

Keys: I

Zoom Out Command

The Zoom Out command reduces the size of the image displayed in the EasyAnalog software main window.

Shortcuts

Keys: O

Settings menu commands

The Settings menu offers the following commands:

Command	Function
Chip settings (see page 31)	**Pop up dialog box to set master clock frequency, clock 0 through clock 3 frequencies, the IBIAS value, the Voltage Reference value, and enable/disable VREF and VMR.**
Serial Port (see page 39)	**Pop up dialog box to allow selection of serial port (COM1 through COM8) that EasyAnalog software will use to send data to the POD (which in turn writes the data to the FPAA chip on the board.)**
Preferences (see page 38)	**Pop up dialog box to allow EasyAnalog software sound effects to be turned on or off.**

Shift Macro Command

This mode is the default mode for the system. It allows you to move a macro about on the screen. You move a macro by placing the cursor over the macro then pressing the left mouse button down and holding it down. The macro can then be moved by moving the mouse. The macro can be placed in its new position by releasing the left mouse button.

NOTE: If you move a macro that has wires attached to it, the wires are deleted. (If you start to move it but change your mind and drop it back where it was originally placed, the wires are not deleted.)

You can exit "Shift Macro" mode by selecting another mode from the "Draw" menu or by pressing one of the following tool-bar buttons:

Shortcuts

Toolbar: 

Keys: Esc

System Minimize Command

Minimize command

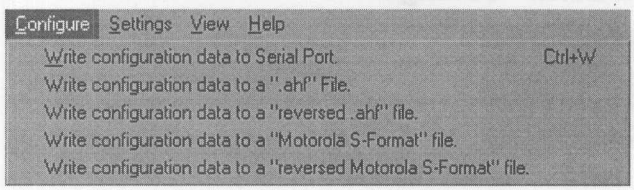
Use this command to reduce the EasyAnalog software window to an icon.

Shortcut

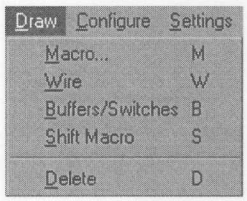
Mouse: Click the minimize icon on the title bar.

Keys: ALT+F9

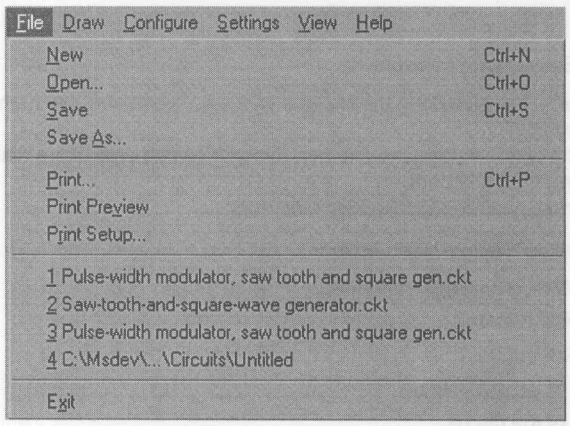
Pull-down - Configure



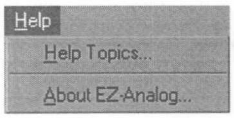
Pull-down - Draw



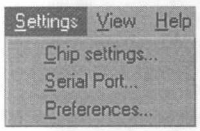
pull-down - file



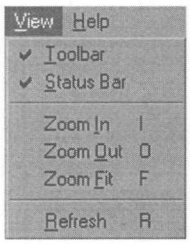
Pull-down - Help



Pull-down - Settings



Pull-down - View



Window Caption

Title Bar

Untitled - EasyAnalog software

The title bar is located along the top of a window. It contains the name of the application and circuit.

To move the window, drag the title bar. Note: You can also move dialog boxes by dragging their title bars.

A title bar may contain the following elements:

- Application Control-menu button
- Document Control-menu button
- Maximize button
- Minimize button
- Name of the application
- Name of the circuit
- Restore button

New Window

New command (Window menu)

Use this command to open a new circuit. You will be prompted to save any unsaved data from the previous circuit before the new circuit image appears. Saved files will have a ".ckt" suffix.

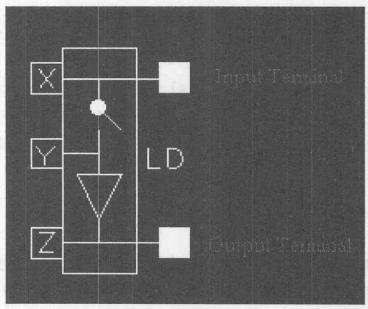
IO Cells

IO Cells

Input and output to the circuitry within the chip must pass through an IO Cell. There are 13 IO Cells around the periphery of the chip. See *IO Cell Details (see page 45)*.

IO Cell Details

IO Cells appear about the periphery of the chip image in the EasyAnalog software main window. Each IO cell has a label such as LD as seen below and external pins X, Y, and Z. There are two internal contacts that may be wired to other objects when in "*wire mode (see page 41)*". Each IO cell also contains a switch that can be toggled on or off (when in "wire mode" or "*delete mode (see page 32)*" using the left mouse button and an opamp that can be toggled on or off using the left mouse button. The IO cell can be used for external input or output. Note that the opamp can also be used to buffer the input or output signals to or from the internal chip circuitry.



IO Cell Output Terminal

Each IO Cell on the chip has a terminal that is designated as an output terminal. ("Output" in this case means output to the IO Cell itself.) See *IO Cell Details (see page 45)*.

IO Cell Identifier

Each IO Cell on the chip has an identifier such as LD, RC or D3 that identifies the corresponding terminal on the board so that you will know how to connect input signals to the chip and monitor output signals from the chip. See *IO Cell Details (see page 45)*.

IO Cell Internal Opamp

Each IO Cell on the chip has an internal opamp that can be turned on or off. (It is turned off by default to save power.) You would turn this opamp on and use it to buffer input to and output from the internal circuitry of the Field-Programmable Gate Array by routing your input and output signals through it. See *IO Cell Details (see page 45)*.

IO Cell Input Terminal

Each IO Cell on the chip has a terminal that is designated as an input terminal. ("Input" in this case means input to the IO Cell itself.) See *IO Cell Details (see page 45)*.

IO Cell Internal Switch

Each IO Cell on the chip has an internal switch that can be used to connect the "X" terminal to the IO Cell opamp. See *IO Cell Details (see page 45)*.

IO Cell X Terminal

Each IO Cell on the chip has an X, Y and Z terminal connected to a labeled pin on the board through which external input signals may be applied or through which output signals from the chip can be transmitted. See *IO Cell Details (see page 45)*.

IO Cell Y Terminal

Each IO Cell on the chip has an X, Y and Z terminal connected to a labeled pin on the board through which external input signals may be applied or through which output signals from the chip can be transmitted. See *IO Cell Details (see page 45)*.

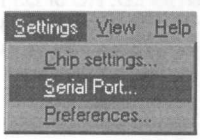
IO Cell Z Terminal

Each IO Cell on the chip has an X, Y and Z terminal connected to a labeled pin on the board through which external input signals may be applied or through which output signals from the chip can be transmitted. See *IO Cell Details (see page 45)*.

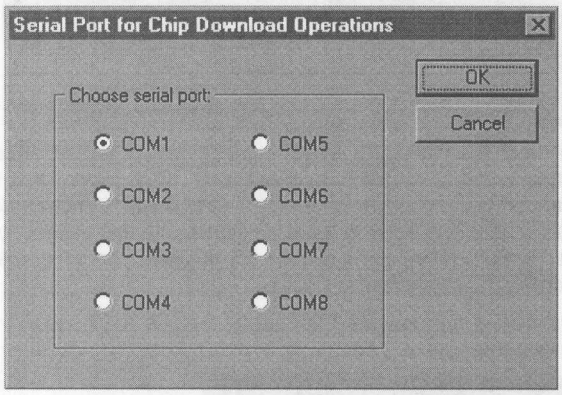
Serial Port

Selecting the Serial Port

From the EasyAnalog software main menu, select the Settings/Serial Port item as shown below:



This will cause the "Serial Port for Chip Download Operations" window to pop up:



Click on the appropriate circle to set the correct Serial Port for your machine.

The correct port is the one to which you have attached *the POD (see page 11)*.

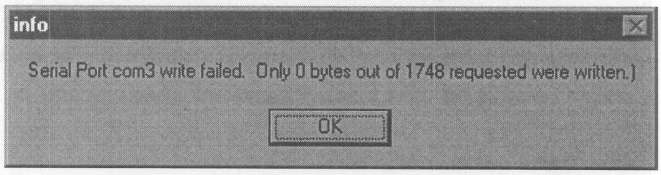
Note that different PC machines have different hardware configurations. Usually the serial port that the POD will be connected to will be COM1 or COM2.

If you are not certain which port is appropriate for your machine please consult the *Serial Port Communications Problems (see page 47)* topic.

Serial Port Communications Problems

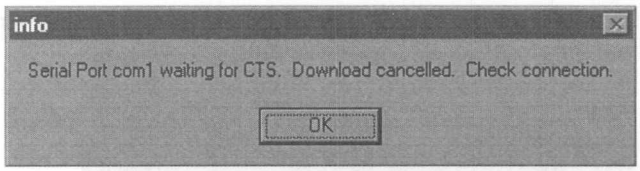
Here is how to run down serial port communications problems AFTER you are sure that you have connected the *Field Programmable Analog Array Hardware (see page 11)*:

1. Run EasyAnalog software and select the File/New menu item so that you have a new "untitled" circuit. (Note that although this circuit does not contain any macros or wires, it may be downloaded to the serial port just as any other circuit, i.e., choosing "Configure/Write configuration data to Serial Port" will cause 1748 bytes of information to be transmitted via the selected serial port to the board/chip via the POD.)
2. Choose the "Configure/Write configuration data to Serial Port" menu item to initiate a configuration download operation.
3. If you see the following:



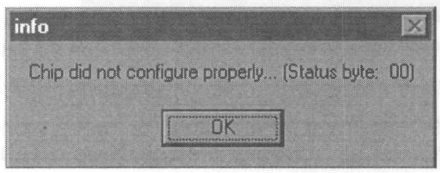
it means that the serial port you selected is not the correct one, i.e., it is either not configured properly or does not exist. You should *select a different serial port (see page 46)*.

4. If you see the following:



it means that the "Clear To Send" signal is not being asserted. No data will be transmitted to the chip until CTS is asserted. CTS means that the device to which the serial port is attached is signaling that it is ready to receive data. *The POD (see page 11)*, if attached, asserts the CTS signal. In this case, you have a valid serial port but *the POD (see page 11)* does not appear to be connected to that port. You should check the connections.

5. If, after initiating the download operation, there is a 13 second delay and you then see the following:



it means that you have a valid serial port and data was sent through it but no data was received back from *the POD (see page 11)*. Normally, the POD sends data back telling whether or not the chip on the board received the data and properly configured itself.

6. If you heard a bell followed by another bell approximately 3 seconds later, the download to the chip was successful and everything should be OK.

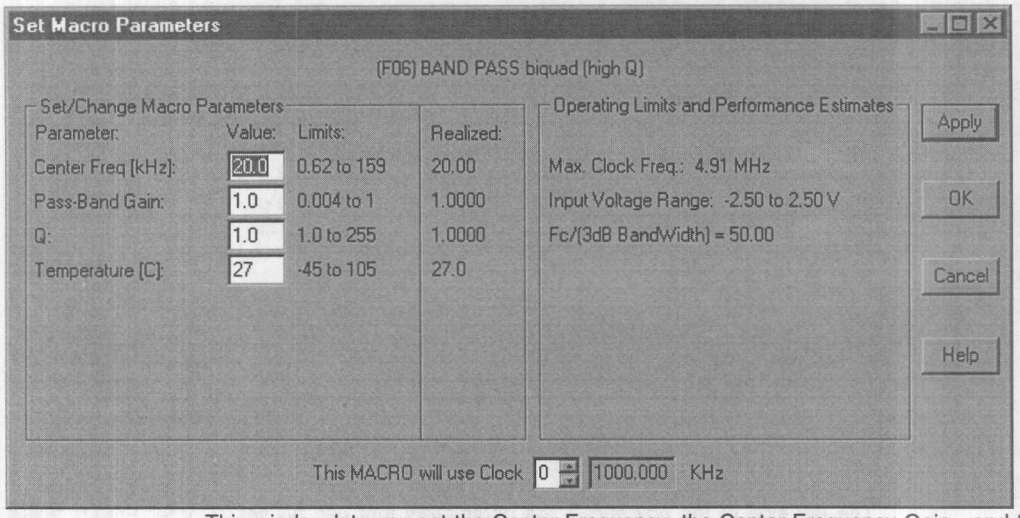
Settings

Setting Macro Parameters

Each macro that has been place on the screen has its own "Set Macro Parameters" window associated with it that allows you to set parameters specific to that instance of that macro. You bring up the "Set Macro Parameters" dialog window by placing the cursor over the image of the macro on the screen and clicking the **right** mouse button.

Each type of macro has its own set of parameters that are relevant to its internal functionality, i.e., some macros have several parameters, some have none.

Shown below is the "Set Macro Parameters" dialog window for the Band Pass BiQuad High Q filter.



This window lets you set the Center Frequency, the Center Frequency Gain, and the Q of the filter in the boxes on the left side of the window. The allowable values for these fields are shown in the "limits" column. The "Realized" column shows the values that were attainable by the system. The "Realized" values will not, in all cases, exactly match your desired values since the internal values are created by selecting discrete capacitor values or discrete capacitor value ratios but will, in most cases, be close to the values you input.

The "Temperature" value should be the temperature at which you expect the FPAA chip to run in you application. It is used in equations that estimate the "Operating Limits and Performance Estimates" that will appear on the right side of the window. These values are only updated when you click on "Apply" or "OK".

Clicking on "OK" actually enters the values into your macro and exits the "Set Macro Parameters" window.

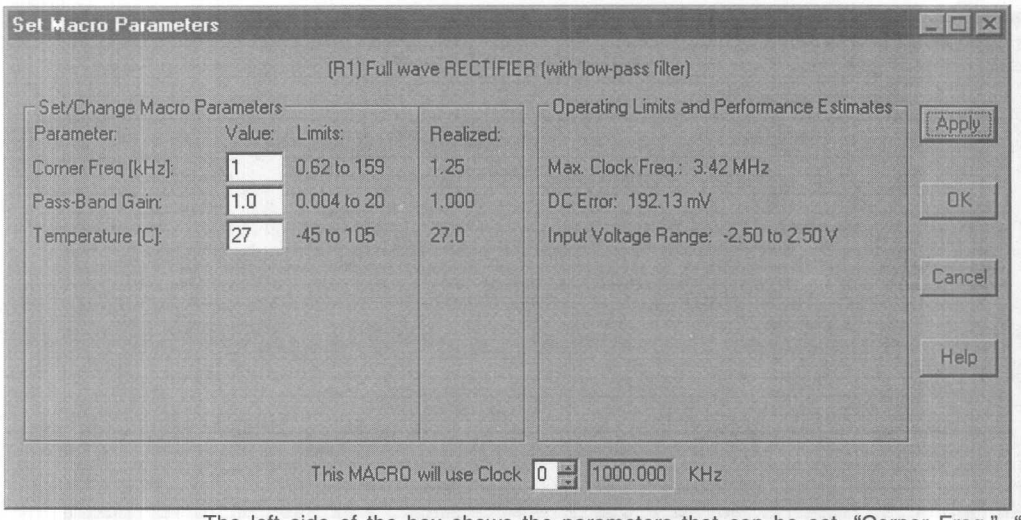
Clicking on Apply only runs the equations and updates the "Operating Limits and Performance Estimates" values so you can see what effect your new parameters would have if you chose to use them.

The "Cancel" button will exit the window and make no changes to the parameter values that existed within the macro when you brought up the "Set Macro Parameters" window.

At the bottom of the macro, is a field that allows you to select which *divide-down clock (see page 53)* will be used by this macro (if it is a macro that indeed uses a clock.) The clock values for the four possible clocks can be set using the "*Set Electrical Parameters (see page 53)*" window.

The Set Macro Parameters pop-up window

The Rectifier pop-up window appears as seen below.



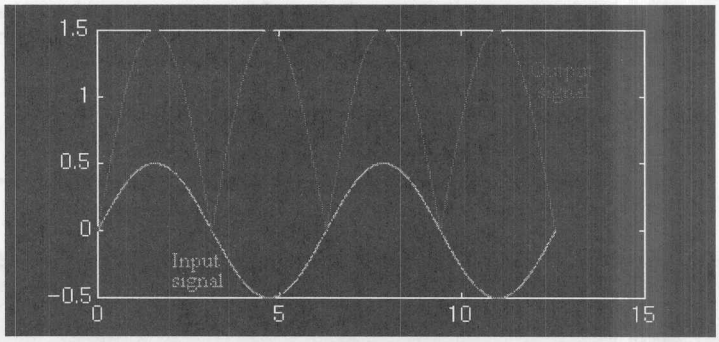
The left side of the box shows the parameters that can be set, "Corner Freq.", "Pass Band Gain", and "Desired Temperature".

Note that this rectifier behaves as if it were a rectifier followed by a gain stage followed by a low-pass filter. The "Corner Frequency" is the frequency at which the output signal is attenuated by 3 dB. So if you set the Corner Frequency to 10 kHz, the output of this rectifier would look like a full amplitude rectified version of the input signal as long as the input signal frequency was rather less than 10 kHz. If the input frequency of the input signal were increased however to 10 kHz, it would be attenuated 3 dB. If the input signal frequency were increased further and further, it would be attenuated more and more (20 dB for every decade). This filtering effect can be put to good use by eliminating harmonics from the output signal when the input signal is below the corner frequency.

The "Limits" column shows the range of acceptable values for a particular parameter. As you can see, the Corner Frequency can range from .62 kHz to 159 kHz. Decimal values such as 2.56 are allowable as input. The "Realized" value will be as close as possible to that value.

The "Realized" column show the value that was achieved by the system. (Recall that the switched-capacitors that we are controlling have 255 discrete values from 1 to 255 units so that the values realized will be close to the value specified but perhaps not exactly the value specified.)

The "Pass Band Gain" refers to the gain of the amplifier in the region unaffected by the low-pass filter, i.e., the signal that is below the corner frequency. If you were to input a 1 Volt peak-to-peak 20 kHz sine wave signal with rectifier macro corner frequency set to 30 kHz and rectifier macro pass band gain set to 3, the output would measure 1.5 volts peak-to-peak as shown below.



Rectification of 1 Volt Peak-to-Peak Sine wave with pass-band gain set to 3.

The "Temperature" setting refers to the Temperature of the environment that you think the chip will run at in your real-world application. The temperature will be used only in EasyAnalog software calculations that compute the "Operating Limits and Performance Estimates".

After entering parameters, you should click on "Apply" to run the Operating Limits and Performance Estimates" calculations. The results of the calculations will be displayed on the right side of the box. "Apply" does NOT change the actual values stored in your macro. (Only "OK" changes the actual parameters within your macro to your newly assigned values.)

If you click on OK, the calculations are run, the new data is assigned to the Macro, and the popup window exits.

If you click on Cancel, the macro values will be unchanged, and the popup window will exit.

The Operating Limits and Performance Estimates" column tells you how your choice of parameters on the left side of the box have affected the operating limits and performance of the circuit you have designed.

For the default corner frequency of 1 MHz the default pass band gain of 1, we see that the Maximum Clock Frequency has been calculated to be 3.42 MHz. This means that this circuit will operate properly at clock frequencies of 3.42 MHz or less. (Caution: you would not want to set the clock frequency to a value less than twice your maximum input signal frequency because "aliasing" will occur. The FPAA chip, because it uses switched-capacitor technology, is a "sampled-data system" so the Nyquist criteria must be met, i.e., the sampling frequency or clock frequency must be at least twice the signal frequency. In practice, the sampling frequency should be at least 10 times the signal frequency if fidelity of the input wave form is to be maintained.)

We also see that the DC Error of the output signal is +/- 192.13 milliVolts, i.e., if the input wave-form has a peak amplitude of 2.0 Volts, the output wave-form peak amplitude should be 2.0 Volts +/- 192.13 milliVolts.

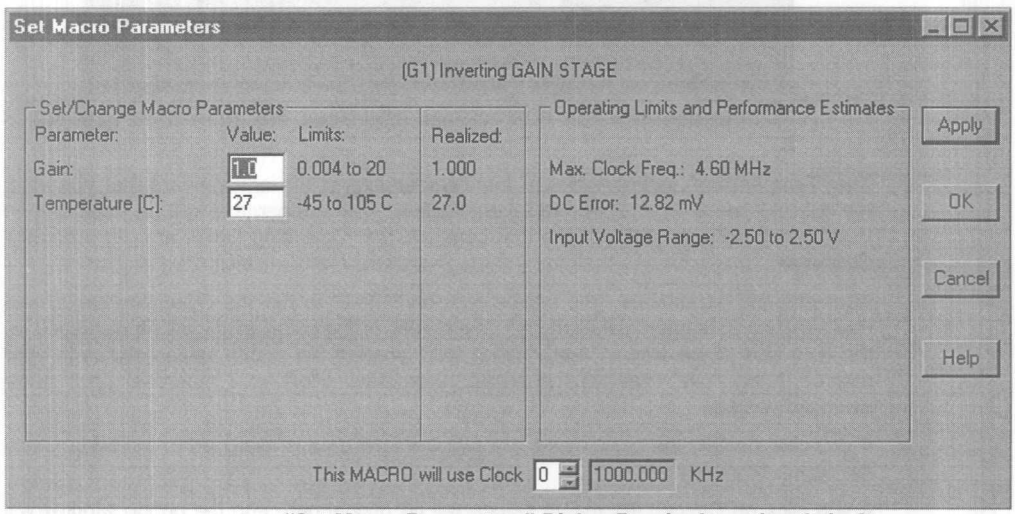
We also see that the input Voltage range should be kept between -2.5 and +2.5 volts otherwise the output signal will be clipped.

The final item in the "Set Macro Parameters" box is the "This MACRO will use Clock . . ." field. There are four internal clocks whose frequencies can be set using the "*Set Electrical Parameters (see page 53)*" menu item. One of the four clocks can be assigned to this particular macro. The clock value is not used by the Rectifier macro but there are

other macros that do make use of the clock value. You should check the documentation on a particular macro to see if it uses the clock value and how the clock value may be used to affect the behavior of that macro.

Setting Macro Parameters for the Gain-Inv macro.

If you put the cursor over the green object labeled "Gain-Inv" and then press the RIGHT mouse button, a pop-up window will appear as seen below.



"Set Macro Parameters" Dialog Box for Inverting Gain Stage.

The left side of the box shows the parameters that can be set, "Desired Gain" and "Desired Temperature".

The "Limits" column shows the range of acceptable values for a particular parameter. As you can see, the gain can range from .004 to 20. Decimal values such as 12.335 are allowable as input. The "Realized" value will be as close as possible to the value you specify.

The "Realized" column show the value that was achieved by the system. (The gain of an opamp is set by the ratio of two switched-capacitors. Each capacitor's value can range from 1 to 255 units. The system judiciously selects capacitor values so that the ratio comes as close as possible to your desired gain value, i.e., the ratio can range from 1/255 to 255/1 units.)

The "Temperature" setting refers to the Temperature of the environment that you think the chip will run at in your real-world application. The temperature will be used only in EasyAnalog software calculations that compute the "Operating Limits and Performance Estimates".

After entering parameters, you should click on "Apply" to run the Operating Limits and Performance Estimates" calculations. The results of the calculations will be displayed on the right side of the box. "Apply" does NOT change the actual values stored in your macro. (Only "OK" changes the actual parameters within your macro to your newly assigned values.)

If you click on OK, the calculations are run, the new data is assigned to the Macro, and the popup window exits.

If you click on Cancel, the macro values will be unchanged, and the popup window will exit.

The Operating Limits and Performance Estimates" column tells you how your choice of parameters on the left side of the box have affected the operating limits and performance of the circuit you have designed.

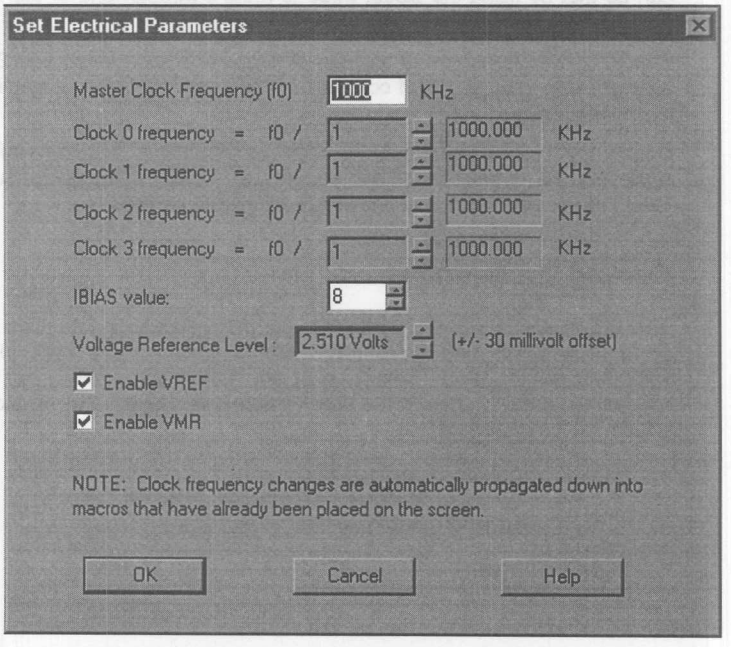
For the default gain of "-1", we see that the Maximum Clock Frequency is 4.60 MHz. This means that this circuit will operate properly at clock frequencies of 4.60 MHz or less.

We also see that the DC Error of the output signal is +/- 12.82 milliVolts, i.e., if the input wave-form has a peak amplitude of 2.0 Volts, the output wave-form peak amplitude should be -2.0 Volts +/- 12.82 milliVolts.

We also see that the input Voltage range should be kept between -2.5 and +2.5 volts.

The final item in the "Set Macro Parameters" box is the "This MACRO will use Clock . . ." field. There are four internal clocks whose frequencies can be set using the "*Set Electrical Parameters" (see page 53)* menu item. One of the four clocks can be assigned to this particular macro. The clock value is not used by the Gain-Inv macro but there are other macros that do make use of the clock value. You should check the documentation on a particular macro to see if it uses the clock value and how the clock value may be used to affect the behavior of that macro.

Setting the Chip's Electrical Parameters



Recall that the Field Programmable Analog Array is based on switched-capacitor circuits and is therefore a "sampled data system". This means that the Master Clock Frequency

sets the sample rate and determines frequency limits for the analog signals being processed by the chip.

The Master Clock Frequency is supplied by you. When you set up the board, you can either use the one megahertz clock on the chip or an external signal source that you provide.

You should set the Master Clock Frequency in the dialog window above to the frequency of the source in any case. The only effect your setting will have is that EasyAnalog software will then be able to properly calculate the Operating Limits and Performance Estimates that appear in the "*Set Macro Parameters (see page 48)*" window associated with each macro.

The four clocks, clock 0, clock 1, clock 2, and clock 3 are derived by dividing-down the master clock frequency. Each can be set to any one of 32 different values. Set the value by left mouse-clicking on either the top half or the lower half of the "spinner" button:

Certain macros in our macro library will make use of one of these clocks. You can cause a macro to use a particular one of these clocks by using the "*Set Macro Parameters (see page 48)*" window associated with that macro.

The IBIAS setting sets the current bias for the opamps in the primary zones (the opamps the macros use.) The nominal value is 8.

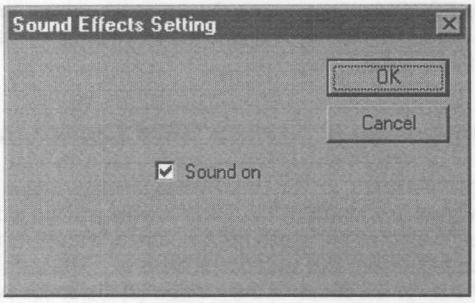
The Voltage Reference is an on-chip voltage source that you can use as a reference voltage. You can set its approximate value typing in a value or using the "spinner" buttons. You should probably measure this voltage to determine its precise value. You can do that by using the global wires to route it through one of the IO Cells to a pin on the board that you have access to.

Enable VREF allows you to turn the *Voltage Reference (see page 28)* on or off. If you are not using it, turning it off could save a little power. Left mouse-click the check box to toggle the setting.

Enable VMR allows you to turn the "Voltage Mid-Range" line on or off. This is essentially signal ground. If you wish to use an external signal ground, you should disable the VMR line. Left mouse-click the check box to toggle the setting.

Setting Preferences

This dialog window allows you to set your preferences for execution of this software. At the moment, the only preference item is whether you wish to have EasyAnalog software's sound effects turned on or off. Clicking the check box will either erase or restore the check mark. If the check mark is displayed, EasyAnalog software's sound effects will be turned on.



Configuration Files

Configuration Data

The term "Configuration Data" refers to the data that is downloaded to the Field Programmable Analog Array chip. This data contains 6864 binary bits. Each bit corresponds to one of the electronic switches within the chip. The electronic switches control the internal circuit connectivity, the capacitor values, voltage reference value, clock settings, etc., within the chip. The EasyAnalog software software allows you to construct circuits using macros, high-level building blocks such as gain stages, rectifiers, oscillators, etc., connect them together, set their parameters, then download the resultant configuration bit-stream to the chip via *the POD (see page 11)*. The chip places the configuration data in its internal registers which are connected to the internal switches which are turned on or off depending on the corresponding binary value in the register.

You can also *download configuration data to a file (see page 32)* so that an EPROM can be programmed. You could then use the EPROM as a source for loading your circuit into the Field-Programmable Analog Array chip.

Write Configuration to a "-r.ahf" File Command

This writes the configuration data, for the circuit being displayed, to a file. The prefix will be the name you assigned to your circuit file, i.e., if the file you are working with is "circuit1.ckt" then the file that will be written will be named "circuit1-r.ahf". The file will be written into the same directory that your circuit file is in. The path name for this file will be displayed in the lower left corner of the window (in the *status bar (see page 39)*) immediately after the file has been written in case there is any question regarding where it has gone.

The "-r.ahf" file contains the ASCII codes for the hexadecimal byte-stream that represents your circuit. The "-r" signifies "reversed" which refers to the fact that the bits within each byte have been reversed to comply with the convention for programming a serial EPROM chip.

The file may be used to program a serial EPROM chip so that your circuit can be downloaded to the Field Programmable Analog Array chip directly from the EPROM.

Note that many of the EPROM programmer software packages are unable to handle names whose prefix is greater than eight characters so you might wish to choose your circuit name so with this in mind.

Write Configuration to a "-r.ms2" File Command

This writes the configuration data, for the circuit being displayed, to a file. The prefix will be the name you assigned to your circuit file, i.e., if the file you are working with is "circuit1.ckt" then the file that will be written will be named "circuit1-r.ms2". The file will be written into the same directory that your circuit file is in. The path name for this file will be displayed in the lower left corner of the window (in the *status bar (see page 39)*) immediately after the file has been written in case there is any question regarding where it has gone.

The "-r.ms2 " file contains the ASCII codes for the hexadecimal byte-stream that represents your circuit.

The file may be used to program a parallel EPROM chip so that your circuit can be downloaded to the Field Programmable Analog Array chip directly from the EPROM.

Write Configuration to a ".ahf" File Command

This writes the configuration data, for the circuit being displayed, to a file. The prefix will be the name you assigned to your circuit file, i.e., if the file you are working with is "circuit1.ckt" then the file that will be written will be named "circuit1.ahf". The file will be written into the same directory that your circuit file is in. The path name for this file will be displayed in the lower left corner of the window (in the *status bar (see page 39)*) immediately after the file has been written in case there is any question regarding where it has gone.

The ".ahf" file contains the ASCII codes for the hexadecimal byte-stream that represents your circuit.

The file may be used to program a parallel EPROM chip so that your circuit can be downloaded to the Field Programmable Analog Array chip directly from the EPROM.

Note that many of the EPROM programmer software packages are unable to handle names whose prefix is greater than eight characters so you might wish to choose your circuit name so with this in mind.

Write Configuration to a ".ms2" File Command

This writes the configuration data, for the circuit being displayed, to a file. The prefix will be the name you assigned to your circuit file, i.e., if the file you are working with is "circuit1.ckt" then the file that will be written will be named "circuit1.ms2". The file will be written into the same directory that your circuit file is in. The path name for this file will be displayed in the lower left corner of the window (in the *status bar (see page 39)*) immediately after the file has been written in case there is any question regarding where it has gone.

The ".ms2" file contains the ASCII codes for the hexadecimal byte-stream that represents your circuit. This file is in "Motorola S-Format" form. This format is recognized by many of the EPROM programmer software packages available from third-parties.

The file may be used to program a parallel EPROM chip so that your circuit can be downloaded to the Field Programmable Analog Array chip directly from the EPROM.

Write Configuration to Serial Port Command

This writes the configuration data, for the circuit being displayed, to the *POD (see page 11)* which in turn writes the configuration data to the Field Programmable Analog Array chip. The PC serial port is connected to the POD so the data is transmitted from the PC to the POD via the serial port. (You need to make sure that you have selected the proper serial port by using the *Settings/Serial Port (see page 46)* menu item.)

As soon as the data has been successfully transmitted (takes approximately 3 seconds), the chip will begin functioning as the circuit that you downloaded. You will be able to see the results on your oscilloscope immediately.

If you experience difficulties with serial download, see the section *on serial port communications problems (see page 47)*.

Shortcuts

Keys: CTRL+W

Macro Reference

Macros are small building-block circuits that can be selected, moved about the screen, placed and wired. You will build your circuits up from the macros in our "circuits.lib" *macro library (see page 60)*.

There are 6864 electronic switches on the chip that can be used to create almost any conceivable analog circuit. Working at the individual switch level can be tedious however so we have created a library of building block circuits whose internal switches have already been set. One of the building blocks, for example, is a gain stage.

There are some 200 switches that we had to choose to set on or off to make one of the chip zones behave as a simple gain stage. Each switch setting Is represented as a single bit. Here are two examples of the hexadecimal (each character represents 4 bits) representations required to express a macro:

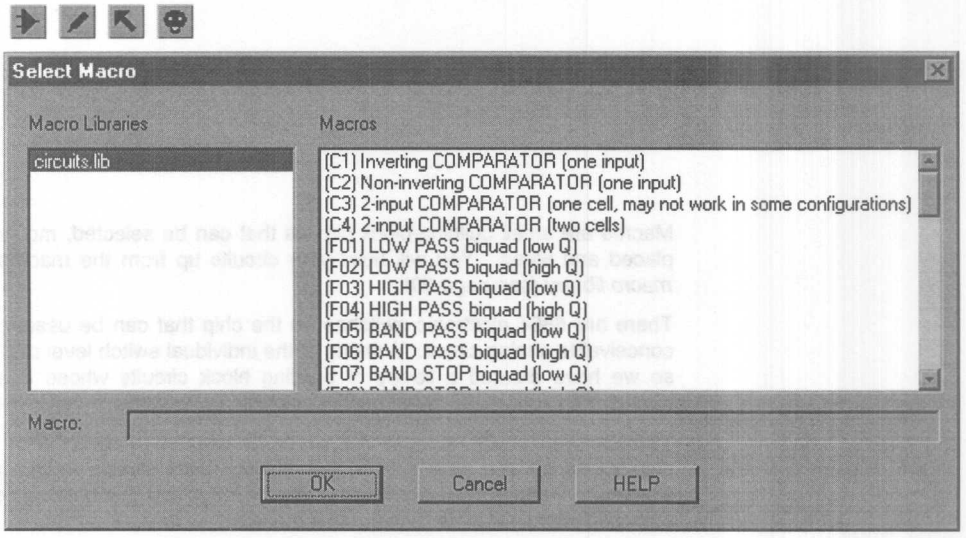
Macro	Bytes
Simple gain stage:	**003f c040 0022 ff24 1000 0ff3 fc00 0018 2270 01c0 0805 2090 8000**
Rectifier:	**003f c040 0082 ff20 1000 0ff0 0000 0080 2a70 01c0 0000 2090 9500**

By creating the *macro library (see page 60)*, circuits.lib, we have made it easy for you. All **you** have to do is use EasyAnalog software to select a macro, place it, wire it up, set its parameters if desired using the "Set Macro Parameters" popup window, download the configuration data to the chip using the "Configure/Write configuration data to serial port" menu item and in approximately 3 seconds, the chip will begin functioning as the circuit you designed. You can use a signal generator and an oscilloscope to verify immediately its operation.

The Select Macro Dialog Box

The select macro dialog box allows you to select a *macro (see page 60)* that you wish to use.

Place the cursor over one of the entries in the "Macro Libraries" column and double-click the left mouse button. The macros contained in that library will be displayed in the "Macros" box on the right. (Circuits.lib is the default library.) You can use the scroll bar on the right to see all of the items in the list. Place the cursor over the name of the macro you wish to use and double-click the left mouse button. (This is the same as clicking once on the item to highlight it then clicking once on the "OK" button.) The macro will be loaded into the program, the Dialog box will disappear, and the macro image will appear in red on the screen. It may be moved about with the mouse. You may place the macro on any of the zone markers by clicking the left mouse button. You may place as many instances of the macro as you wish until you run out of empty chip zones. You can exit "place" mode by pressing the escape key (Esc) or by selecting another mode from the "Draw" menu or by pressing one of the following tool-bar buttons :



The Select Macro Dialog Box

Macros in Circuits.lib Macro Library

A macro is a small building-block circuit that can be selected, moved about the screen, placed and wired. You will build your circuits up from the macros in our "circuits.lib" library. The "circuits.lib" library currently contains the macros listed below.

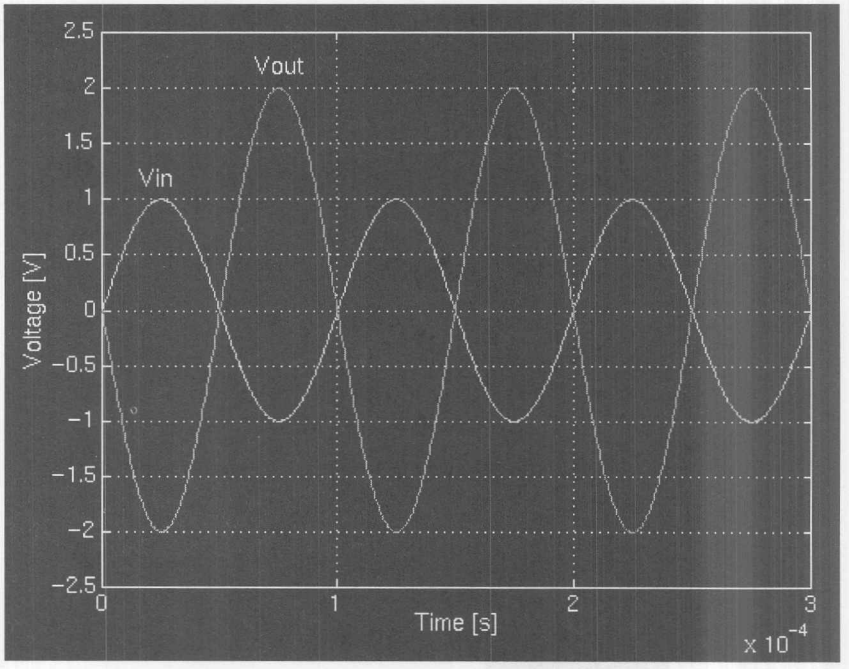
Gain Stages

(G4) Inverting GAIN STAGE (offset comp. in one phase) (see page 67)

(G6) Inverting SUM AMPLIFIER (see page 69)

(G1) Inverting GAIN STAGE

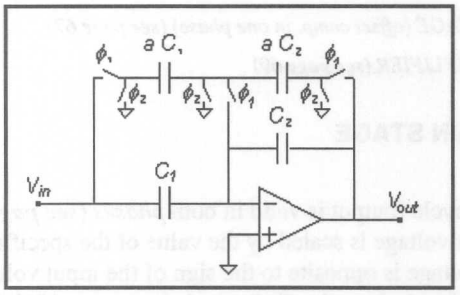
This macro is a full cycle (output is valid in both *phases (see page 150)*) inverting gain stage. The input voltage is scaled by the value of the specified gain, G. The sign of the output voltage is opposite to the sign of the input voltage. An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit is

$$\frac{V_{out}}{V_{in}} = -\frac{C_1}{C_2} = -G$$

where C_1 and C_2 are the input and feedback capacitors respectively as depicted in the figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The constant *a* is not programmable.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Gain	
Description:	voltage gain, *G*
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

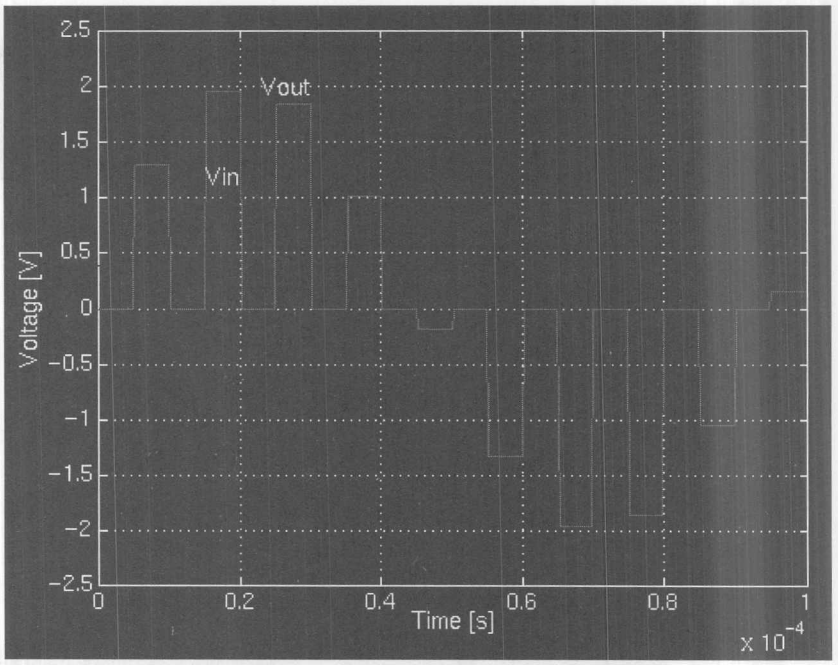
Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$G = \frac{C_1}{C_2}$$

(G2) Half-cycle GAIN STAGE (with offset comp.)

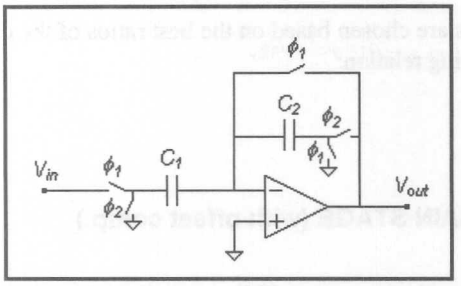
This macro is a half cycle (output is valid in one *phase (see page 150)*) non-inverting gain stage. The input voltage is scaled by the value of the specified gain, G. The sign of the output voltage is the same as the sign of the input voltage (non-inverting gain stage). An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit in phase 2 is

$$\frac{V_{out}}{V_{in}} = \frac{C_1}{C_2} = G$$

where C_1 and C_2 are the input and feedback capacitors respectively as depicted in the figure:



The symbols ϕ_1 and ϕ_2 specify the clock phases. The advantage of this circuit over the gain stage *G1 (see page 61)* is that the output of this circuit is offset compensated. In order to produce output valid in both phases output of this circuit should be conditioned with the sample-and-hold circuit *S1 (see page 72)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	no
Output valid:	no (Vout=0)	yes (offset compensated)

Programmable parameters

Gain	
Description:	voltage gain of the gain stage, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

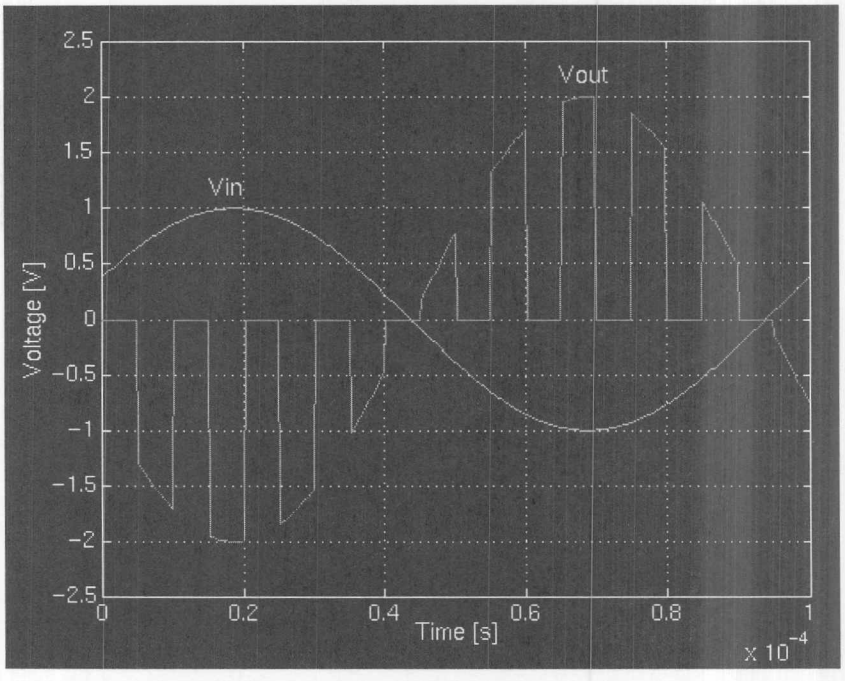
Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$G = \frac{C_1}{C_2}$$

(G3) Half-cycle inverting GAIN STAGE (with offset comp.)

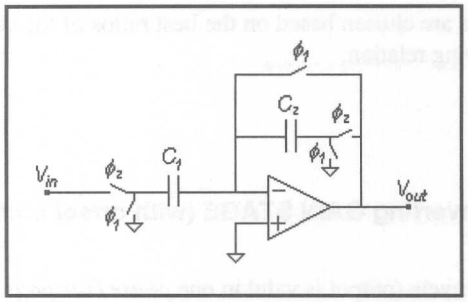
This macro is a half cycle (output is valid in one *phase (see page 150)*) inverting gain stage. The input voltage is scaled by the value of the specified gain, G. The sign of the output voltage is opposite to the sign of the input voltage (inverting gain stage). An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit in phase 2 is

$$\frac{V_{out}}{V_{in}} = -\frac{C_1}{C_2} = -G$$

where C_1 and C_2 are the input and feedback capacitors respectively as depicted in the figure



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The advantage of this circuit over the gain stage *G1 (see page 61)* is that the output of this circuit is offset compensated. In order to produce output valid in both phases output of this circuit should be conditioned with the sample-and-hold circuit *S1 (see page 72)* or track-and-hold circuit *S2 (see page 74)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	no (Vout=0)	yes (offset compensated)

Programmable parameters

Gain	
Description:	voltage gain of the gain stage, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

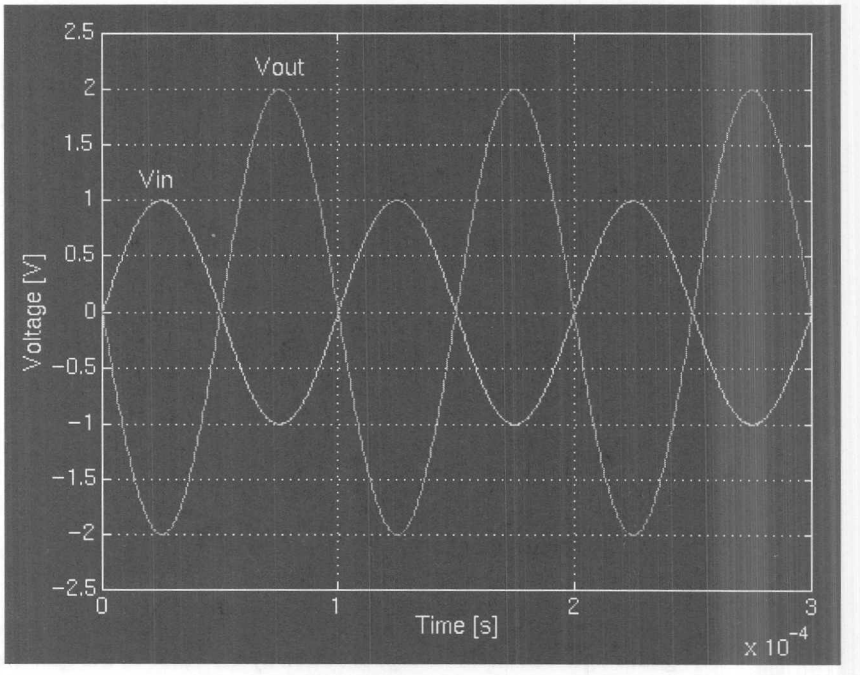
Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$G = \frac{C_1}{C_2}$$

(G4) Inverting GAIN STAGE (offset compensated in one phase)

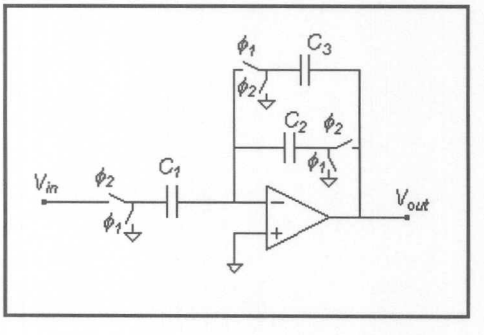
This macro is a full cycle (output is valid in both *phases (see page 150)*) inverting gain stage. The input voltage is scaled by the value of the specified gain, G. The sign of the output voltage is opposite to the sign of the input voltage. In phase 2 the opamp's offset is compensated. An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit in phase 2 is

$$\frac{V_{out}}{V_{in}} = -\frac{C_1}{C_2} = -G$$

where C_1 and C_2 are the input and feedback capacitors respectively as depicted in the figure



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The advantage of this circuit over the gain stage *G1 (see page 61)* is that the output of this circuit is offset compensated in one phase (phase 2).

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes (offset compensated)

Programmable parameters

Gain	
Description:	voltage gain of the gain stage, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

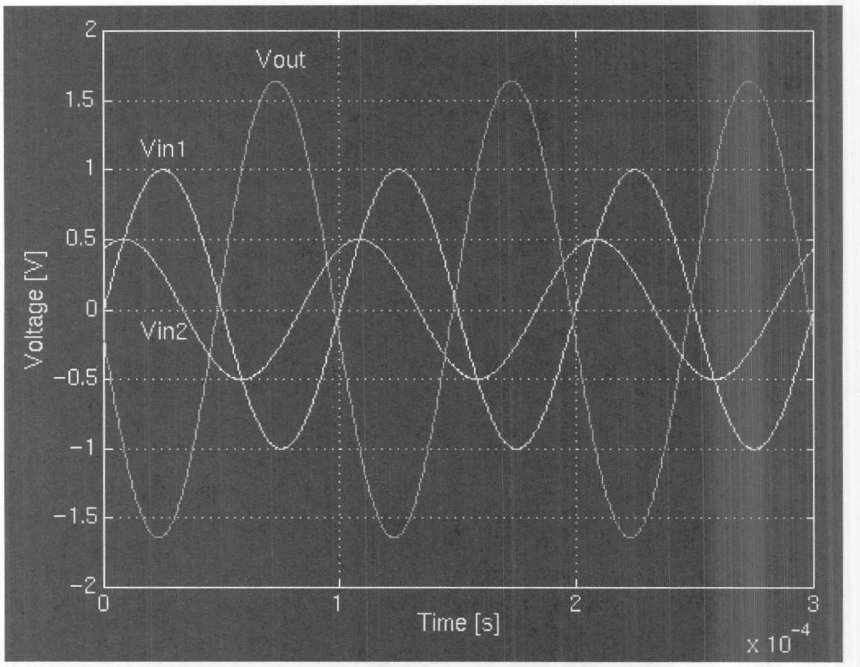
Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$G = \frac{C_1}{C_2}$$

(G6) Inverting SUM AMPLIFIER

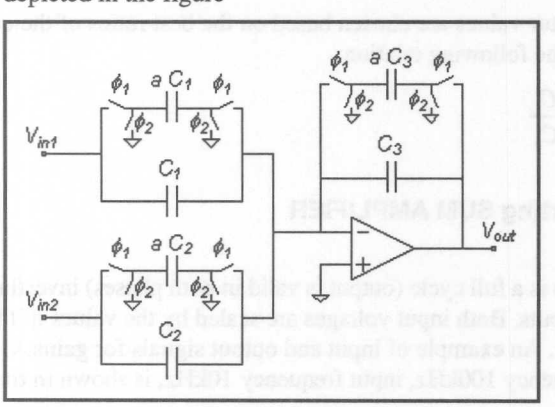
This macro is a full cycle (output is valid in both phases) inverting sum amplifier. It has two inputs. Both input voltages are scaled by the values of the specified gains, G1 and G2. An example of input and output signals for gains, G1=1, G2=0.5, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit is

$$V_{out} = -\frac{C_1}{C_3}V_{in1} - \frac{C_2}{C_3}V_{in2}$$
$$= -G_1V_{in1} - G_2V_{in2}$$

where C_1 and C_2 are the input capacitors, and C_3 is the feedback capacitor as depicted in the figure



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The constant *a* is not programmable and is set to 1.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Gain1 (top)	
Description:	voltage gain of the top input, G_1
Range:	0.004 to 4.0
Default value:	1.0
Unit:	V/V

Gain2 (bottom)	
Description:	voltage gain of the bottom input, G_2
Range:	0.004 to 4.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

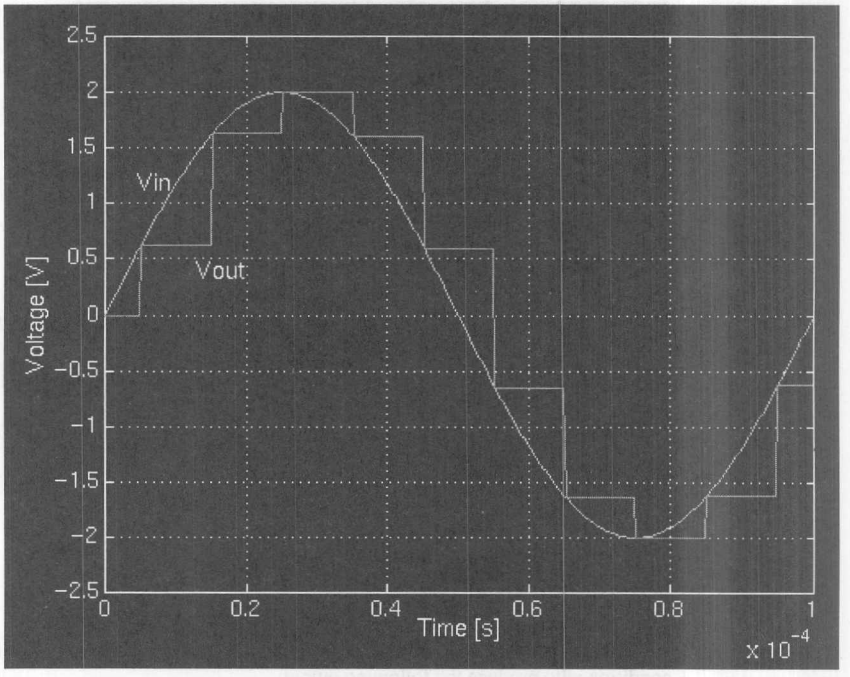
$$G_1 = \frac{C_1}{C_3}$$

$$G_2 = \frac{C_2}{C_3}$$

Signal Conditioning

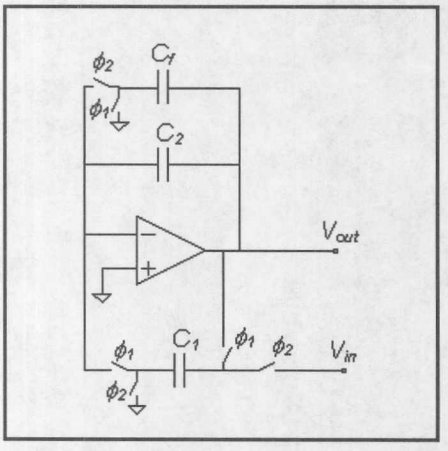
(S1) SAMPLE-and-hold (half cycle offset compensated)

This macro is a sample-and-hold stage with the opamp's offset compensation in phase 2. The input waveform is conditioned to produce the following output:



Some macros produce output that is valid only in phase 2. This circuit may be used to create a valid signal in phase 1 by holding the voltage value at the end of phase 2. It may also be used to change the track-type signal in the phase 2 to a hold-type signal.

Schematic of this circuit is presented in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

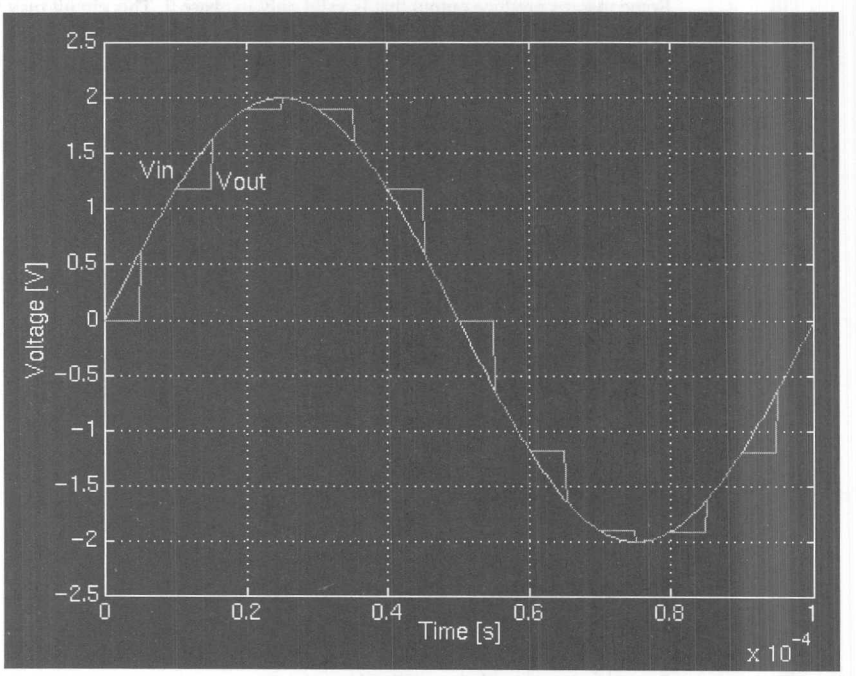
	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes (offset compensated)

Programmable parameters

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

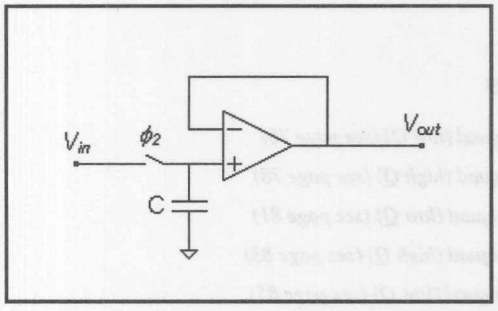
(S2) TRACK-and-hold

This macro is a track-and-hold stage. The input is sampled in phase 1. The input waveform is conditioned to produce the following output:



Some macros produce output that is valid only in phase 2. This circuit may be used to create a valid signal in phase 1 by holding the voltage value at a constant level sampled at the end of phase 2. If the preservation of the track-type signal in phase 2 is not crucial than the macro *S1 (see page 72)* may be used instead (it also provides offset compensation of the opamp).

The schematic of this circuit is presented in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Filter Stages

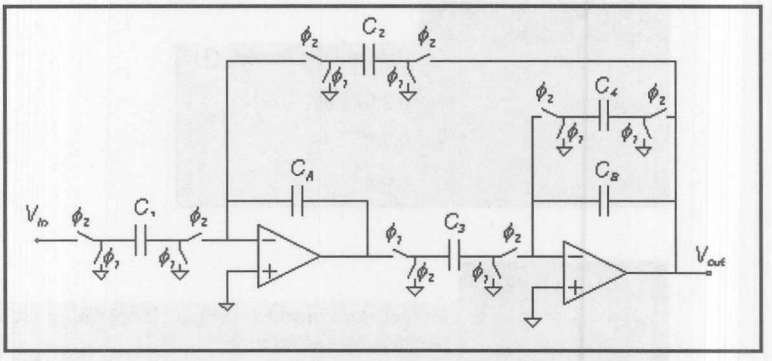
(F1) LOW PASS biquad (low Q)

This macro is a full cycle low pass filter based on biquadratic transfer function. It is designed for low Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{4\pi^2 f_0^2 G}{s^2 + \frac{2\pi f_0}{Q} s + 4\pi^2 f_0^2}$$

where G is the pass-band gain (DC gain), f_0 is the corner frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
Description:	corner frequency, f_0 (frequency at which the gain is GQ [V/V], where G is the pass band gain, and Q is the quality factor)
Range:	8.0 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	20.0
Unit:	kHz

Pass Band Gain	
Description:	voltage pass-band gain, G (DC gain)
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Q	
Description:	**pole quality factor, Q**
Range:	**0.5 to 1.0**
Default value:	**0.707**
Unit:	**none**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**˚C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a low pass filter with 0.3dB ripple, 10kHz pass frequency edge, 50kHz stop frequency edge, and 60dB attenuation using Chebyshev approximation, the following cells are used:

Filter - Low Pass BQ - Low Q: Q=0.668, f0=6.535 kHz, G=0.983

Filter - Low Pass BQ - High Q: Q=2.628, f0=10.652 kHz, G=0.983

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_2}$$

$$Q = \frac{C_3}{C_4}$$

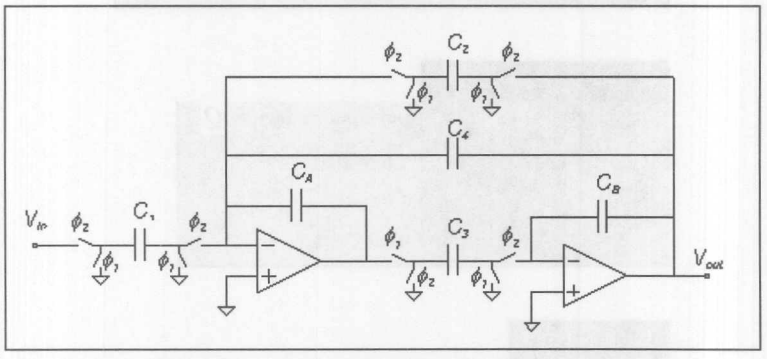
(F2) LOW PASS biquad (high Q)

This macro is a full cycle low pass filter based on biquadratic transfer function. It is designed for high Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{4\pi^2 f_0^2 G}{s^2 + \frac{2\pi f_0}{Q} s + 4\pi^2 f_0^2}$$

where G is the pass-band gain (high frequency gain), f_0 is the corner frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
Description:	**corner frequency, f_0 (frequency at which the gain is GQ [V/V], where G is the pass band gain, and Q is the quality factor)**
Range:	**10 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)**
Default value:	**20.0**
Unit:	**kHz**

Pass-Band Gain	
Description:	**voltage pass-band gain, G (DC gain)**
Range:	**0.004 to 20.0**
Default value:	**1.0**
Unit:	**V/V**

Q	
Description:	**pole quality factor, Q**
Range:	**1.0 to 20.0**
Default value:	**1.0**
Unit:	**none**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a low pass filter with 0.3dB ripple, 10kHz pass frequency edge, 50kHz stop frequency edge, and 60dB attenuation using Chebyshev approximation, the following cells are used:

Filter - Low Pass BQ - Low Q: Q=0.668, f0=6.535 kHz, G=0.983

Filter - Low Pass BQ - High Q: Q=2.628, f0=10.652 kHz, G=0.983

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_2}$$

$$Q = \frac{C_A}{C_4}$$

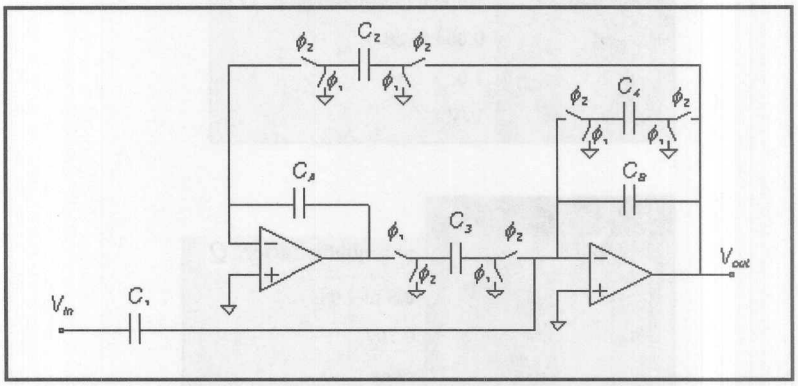
(F3) HIGH PASS biquad (low Q)

This macro is a full cycle high pass filter based on biquadratic transfer function. It is designed for low Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{Gs^2}{s^2 + \frac{2\pi f_0}{Q}s + 4\pi^2 f_0}$$

where G is the pass-band gain (high frequency gain), f_0 is the corner frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	no	yes

Programmable parameters

Corner Frequency	
Description:	**corner frequency, f_0 (frequency at which the gain is GQ [V/V], where G is the pass band gain, and Q is the quality factor)**
Range:	**8.0 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)**
Default value:	**20.0**
Unit:	**kHz**

Pass-Band Gain	
Description:	**voltage pass-band gain, G**
Range:	**0.004 to 20.0**
Default value:	**1.0**
Unit:	**V/V**

Q	
Description:	**pole quality factor, Q**
Range:	**0.5 to 1.0**
Default value:	**0.707**
Unit:	**none**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a high pass filter with 0.3dB ripple, 10kHz stop frequency edge, 50kHz pass frequency edge, and 60dB attenuation using Chebyshev approximation, the following cells are used:

Filter - High Pass BQ - Low Q: Q=0.668, f0=77.17 kHz, G=0.983

Filter - High Pass BQ - High Q: Q=2.628, f0=47.35 kHz, G=0.983

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_B}$$

$$Q = \frac{C_3}{C_4}$$

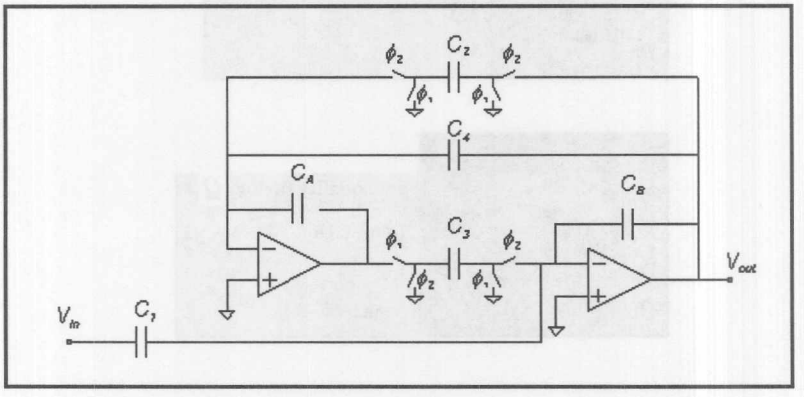
(F4) HIGH PASS biquad (high Q)

This macro is a full cycle high pass filter based on biquadratic transfer function. It is designed for high Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{Gs^2}{s^2 + \frac{2\pi f_0}{Q}s + 4\pi^2 f_0}$$

where G is the pass-band gain (high frequency gain), f_0 is the corner frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	no	yes

Programmable parameters

Corner Frequency	
Description:	corner frequency, f_0 (frequency at which the gain is GQ [V/V], where G is the pass band gain, and Q is the quality factor)
Range:	8.0 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	20.0
Unit:	kHz

Pass-Band Gain	
Description:	voltage pass-band gain, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Q	
Description:	pole quality factor, Q
Range:	1.0 to 20.0
Default value:	1.0
Unit:	none

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**˚C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a high pass filter with 0.3dB ripple, 10kHz stop frequency edge, 50kHz pass frequency edge, and 60dB attenuation using Chebyshev approximation, the following cells are used:

Filter - High Pass BQ - Low Q: Q=0.668, f0=77.17 kHz, G=0.983

Filter - High Pass BQ - High Q: Q=2.628, f0=47.35 kHz, G=0.983

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_B}$$

$$Q = \frac{C_A}{C_4}$$

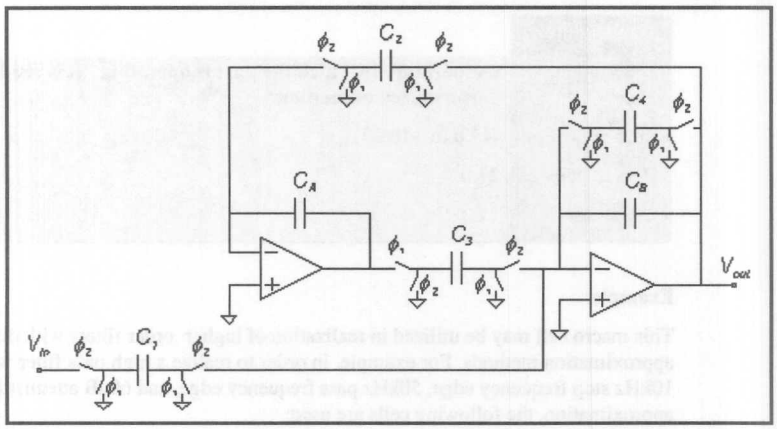
(F5) BAND PASS biquad (low Q)

This macro is a full cycle band pass filter based on biquadratic transfer function. It is designed for low Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{2\pi f_0 \frac{G}{Q} s}{s^2 + \frac{2\pi f_0}{Q} s + 4\pi^2 f_0^2}$$

where G is the pass-band gain (center frequency gain), f_0 is the center frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Center Frequency	
Description:	**center frequency, f_0**
Range:	**8.0 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)**
Default value:	**20.0**
Unit:	**kHz**

Pass-Band Gain	
Description:	**voltage pass-band gain, G**
Range:	**0.004 to 20.0**
Default value:	**1.0**
Unit:	**V/V**

Q	
Description:	**pole quality factor, Q**
Range:	**0.5 to 1.0**
Default value:	**0.707**
Unit:	**none**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a band pass filter with 0.5dB ripple, 3kHz and 10kHz pass frequency edges, 1kHz and 30kHz stop frequency edges, and 40dB attenuation using Butterworth approximation, the following cells are used:

Filter - Band Pass BQ - High Q: Q=2.00578 G=1.61162

Filter - Band Pass BQ - High Q: Q=2.00578 G=6.90264

Filter - Band Pass BQ - Low Q: Q=0.71284 G=0.76894

Filter - Band Pass BQ - Low Q: Q=0.71284 G=1.82724

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_4}$$

$$Q = \frac{C_3}{C_4}$$

Derived Parameters

$$3\text{-dB bandwidth} = \frac{\omega_0}{Q} = f_c\frac{C_4}{C_B} = f_c\frac{C_2C_4}{C_AC_3}$$

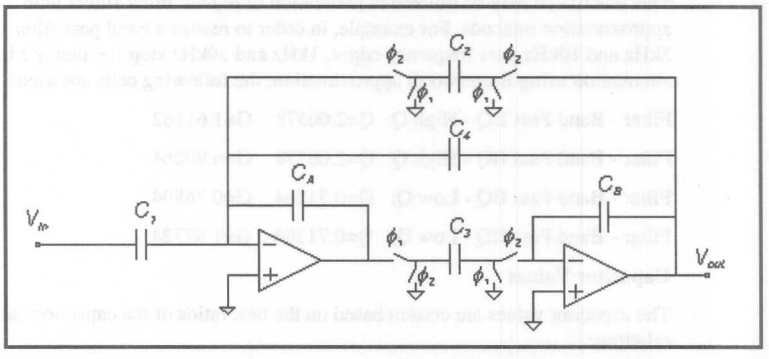
(F6) BAND PASS biquad (high Q)

This macro is a full cycle band pass filter based on biquadratic transfer function. It is designed for high Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{2\pi f_0\frac{G}{Q}s}{s^2 + \frac{2\pi f_0}{Q}s + 4\pi^2 f_0^2}$$

where G is the pass-band gain (center frequency gain), f_0 is the center frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	no	yes

Programmable parameters

Center Frequency	
Description:	center frequency, f_0
Range:	4.0 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	20.0
Unit:	kHz

Pass-Band Gain	
Description:	voltage pass-band gain, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Q	
Description:	pole quality factor, Q
Range:	1.0 to 20.0
Default value:	1.0
Unit:	none

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a band pass filter with 0.5dB ripple, 3kHz and 10kHz pass frequency edges, 1kHz and 30kHz stop frequency edges, and 40dB attenuation using Butterworth approximation, the following cells are used:

Filter - Band Pass BQ - High Q: Q=2.00578 G=1.61162

Filter - Band Pass BQ - High Q: Q=2.00578 G=6.90264

Filter - Band Pass BQ - Low Q: Q=0.71284 G=0.76894

Filter - Band Pass BQ - Low Q: Q=0.71284 G=1.82724

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_4}$$

$$Q = \frac{C_A}{C_4}$$

Derived Parameters

$$3\text{-dB bandwidth} = \frac{\omega_0}{Q} = f_c\frac{C_2C_4}{C_A^2} = f_c\frac{C_3C_4}{C_AC_B}$$

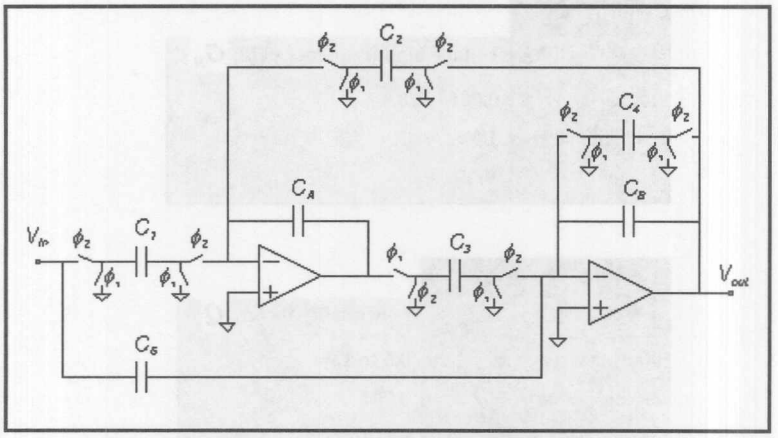
(F7) BAND STOP biquad (low Q)

This macro is a full cycle band stop filter based on biquadratic transfer function. It is designed for low Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{G_H s^2 + 4\pi^2 f_0^2 G_L}{s^2 + \frac{2\pi f_0}{Q}s + 4\pi^2 f_0^2}$$

where G_L is the low frequency pass-band gain, G_H is the high frequency pass-band gain, f_0 is the center frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.**Input-output sampling specifications**

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	no	yes

Programmable parameters

Center Frequency	
Description:	**center frequency, f_0**
Range:	**10 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)**
Default value:	**20.0**
Unit:	**kHz**

GainLO	
Description:	**voltage low frequency (DC) gain, G_L**
Range:	**0.004 to 20.0**
Default value:	**1.0**
Unit:	**V/V**

GainHI	
Description:	**voltage high frequency gain,** G_H
Range:	**0.004 to 20.0**
Default value:	**1.0**
Unit:	**V/V**

Q	
Description:	**pole quality factor,** Q
Range:	**0.5 to 1.0**
Default value:	**0.707**
Unit:	**none**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a band pass filter with 0.1dB ripple, 1kHz and 20kHz pass frequency edges, 2kHz and 10kHz stop frequency edges, and 40dB attenuation using elliptic approximation, the following cells are used:

Filter - Band Stop BQ - Low Q: Q=0.96 f0=7.48 Glo=10.34 Ghi=0.25

Filter - Band Stop BQ - Low Q: Q=0.96 f0=2.68 Glo=0.006 Ghi=0.25

Filter - Band Stop BQ - High Q: Q=3.44 f0=10.9 GL=0.962 GH=0.25

Filter - Band Stop BQ - High Q: Q=3.44 f0=1.84 GL=0.065 GH=0.25

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G_L = \frac{C_1}{C_2}$$

$$G_H = \frac{C_5}{C_B}$$

$$Q = \frac{C_3}{C_4}$$

Derived Parameters

$$\text{Notch frequency} = 2\pi f_0\sqrt{\frac{G_L}{G_H}} = 2\pi f_0\sqrt{\frac{C_1 C_B}{C_2 C_5}}$$

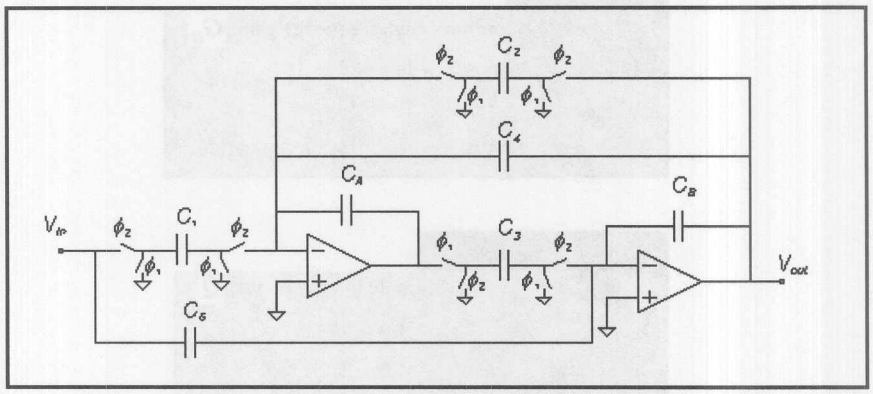
(F8) BAND STOP biquad (high Q)

This macro is a full cycle band stop filter based on biquadratic transfer function. It is designed for high Q (quality factor) values.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{G_H s^2 + 4\pi^2 f_0^2 G_L}{s^2 + \frac{2\pi f_0}{Q}s + 4\pi^2 f_0^2}$$

where G_L is the low frequency pass-band gain, G_H is the high frequency pass-band gain, f_0 is the center frequency, and Q is the quality factor. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	no	yes

Programmable parameters

Center Frequency	
Description:	center frequency, f_0
Range:	4.0 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	20.0
Unit:	kHz

GainLO	
Description:	voltage low frequency (DC) gain, G_L
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

GainHI	
Description:	voltage high frequency gain, G_H
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Q	
Description:	pole quality factor, Q
Range:	1.0 to 255.0
Default value:	1.0
Unit:	none

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Example:

This macro cell may be utilized in realization of higher order filters with different approximation methods. For example, in order to realize a band pass filter with 0.5dB ripple, 3kHz and 10kHz pass frequency edges, 1kHz and 30kHz stop frequency edges, and 40dB attenuation using Butterworth approximation, the following cells are used:

Filter - Band Pass BQ - High Q: Q=2.00578 G=1.61162

Filter - Band Pass BQ - High Q: Q=2.00578 G=6.90264

Filter - Band Pass BQ - Low Q: Q=0.71284 G=0.76894

Filter - Band Pass BQ - Low Q: Q=0.71284 G=1.82724

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$G_L = \frac{C_1}{C_2}$$

$$G_H = \frac{C_5}{C_B}$$

$$Q = \frac{C_A}{C_4}$$

Derived Parameters

$$\text{Notch frequency} = 2\pi f_0\sqrt{\frac{G_L}{G_H}} = 2\pi f_0\sqrt{\frac{C_1 C_B}{C_2 C_5}}$$

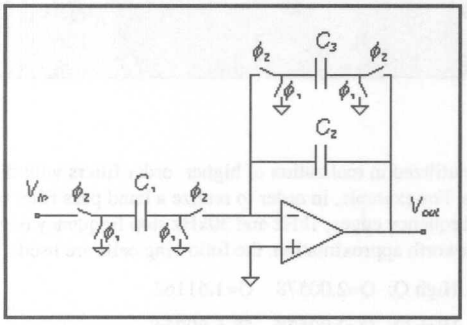
(F9) LOW PASS single pole filter

This macro is a full cycle low pass single pole filter.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{2\pi f_0 G}{s + 2\pi f_0}$$

where G is the pass-band gain and f_0 is the corner frequency. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
Description:	corner frequency, f_0 (frequency at which the gain is $-3+20\log G$ dB, where G is the pass band gain)
Range:	10 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	20.0
Unit:	kHz

Pass-Band Gain	
Description:	**voltage gain (DC gain), G**
Range:	**0.004 to 20.0**
Default value:	**1.0**
Unit:	**V/V**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_3}{C_2}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_3}$$

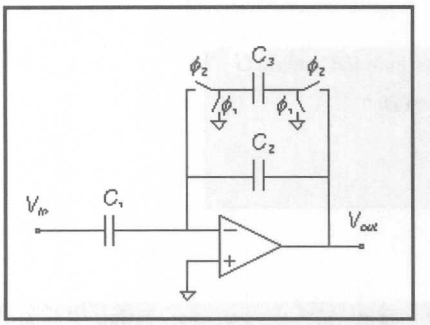
(F10) HIGH PASS single pole filter

This macro is a full cycle high pass single pole filter.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{Gs}{s+2\pi f_0}$$

where G is the pass-band gain and f_0 is the corner frequency. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	no	yes

Programmable parameters

Corner Frequency	
Description:	corner frequency, f_0 (frequency at which the gain is $-3+20\log G$ dB, where G is the pass band gain)
Range:	10 to 100 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	20.0
Unit:	kHz

Pass-Band Gain	
Description:	voltage gain (DC gain), G
Range:	0.004 to 4.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_3}{C_2}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_2}$$

(F11) COSINE filter

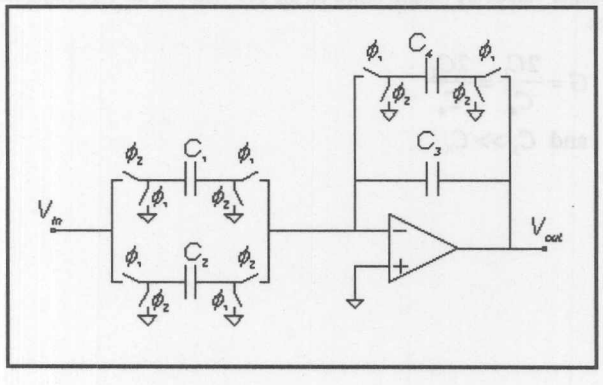
This macro is a full cycle cosine filter stage. It is a special case of a decimator (a low pass filter clocked at a higher harmonic nf_c for which n=2).

The magnitude response of this circuit is

$$\left|\frac{V_{out}}{V_{in}}\right| = G\left|\cos\frac{\pi f}{f_c}\right|$$

where f and f_c are input and clock frequencies respectively.

The circuit is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

DC Gain	
Description:	DC voltage gain, G
Range:	0.2 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$G = \frac{2C_1}{C_4} = \frac{2C_2}{C_4}$$

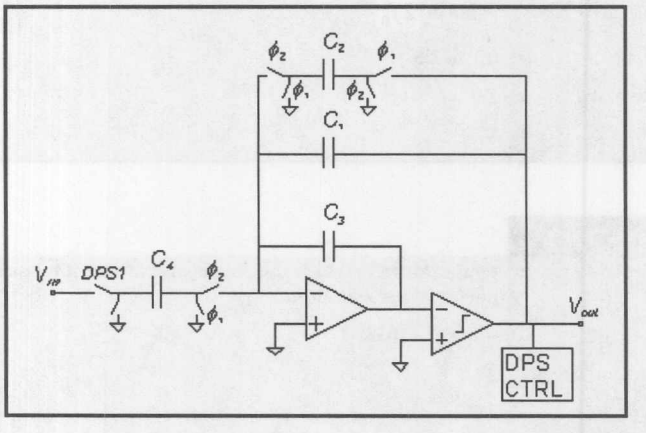
and $C_3 >> C_4$

Oscillators

(O1) Square wave voltage controlled OSCILLATOR

This macro is a voltage controlled square wave oscillator. The higher the input voltage the higher the oscillation frequency.

The schematic of this circuit is presented in the following figure:



The symbols ϕ_1 , ϕ_2 , and $DPS1$ specify the *clock phases (see page 150)*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	if Vin<0
Output valid:	yes	yes

Programmable parameters

Osc. Freq.	
Description:	oscillation frequency (output frequency for zero input voltage)
Range:	0.98 to 250 for the default clock frequency of 1MHz (for other frequencies the range has to be divided by the ratio of 1000/Fc, where fc is the clock frequency in kHz)
Default value:	10.0
Unit:	kHz

Max Vin	
Description:	largest value of the input voltage for which the output frequency has the value of $2f_0$
Range:	0.1 to 2.5
Default value:	1.0
Unit:	V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f = \frac{f_c}{4}\frac{C_2}{C_1} + \frac{f_c}{10}\frac{C_3 C_4}{C_1} V_{in}$$

$$f_0 = \frac{f_c}{4}\frac{C_2}{C_1}$$

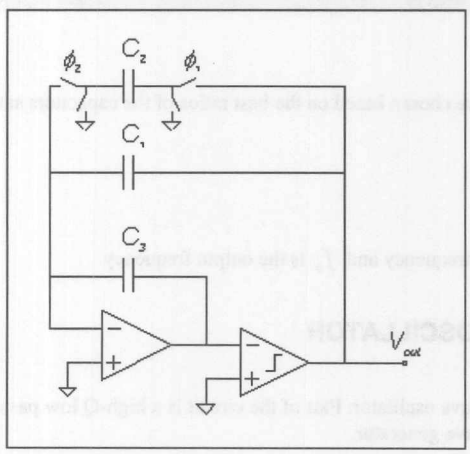
$$f_0 = \frac{f_c}{10}\frac{C_3 C_4}{C_1} V_{\max}$$

where f_c is the clock frequency, f_o is the output frequency when the input voltage is zero, and $V_{\max}$ is the maximum value of the input voltage for which the output frequency is $2f_0$. When the input voltage is $-V_{\max}$ then the output frequency is zero.

(O2) Square wave OSCILLATOR

This macro is a square wave oscillator.

The schematic of this circuit is presented in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	no
Output valid:	yes	yes

Programmable parameters

Osc. Freq.	
Description:	**oscillation frequency**
Range:	**0.98 to 250 for the default clock frequency of 1MHz (for other frequencies the range has to be divided by the ratio 1000/fc, where fc is the clock frequency in kHz)**
Default value:	**10.0**
Unit:	**kHz**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**˚C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

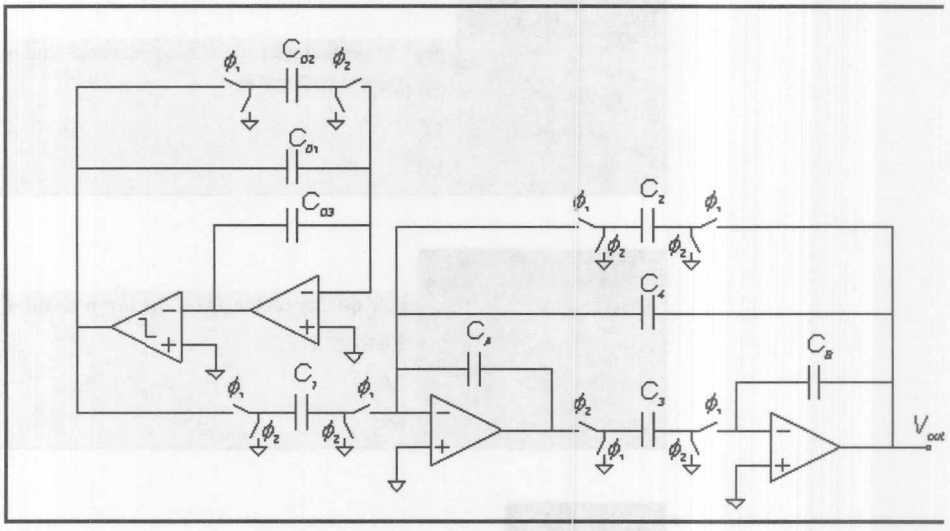
$$f_0 = \frac{f_c}{4}\frac{C_2}{C_1}$$

where f_c is the clock frequency and f_o is the output frequency.

(O3) Sine wave OSCILLATOR

This macro is a sine wave oscillator. Part of the circuit is a high-Q low pass filter which takes the input from a square wave generator.

The schematic of this circuit is presented in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	no
Output valid:	yes	yes

Programmable parameters

Osc. Freq.	
Description:	**oscillation frequency**
Range:	**0.98 to 250 for the default clock frequency of 1MHz (for other frequencies the range has to be divided by the ratio 1000/fc, where fc is the clock frequency in kHz)**
Default value:	**10.0**
Unit:	**kHz**

Amplitude (pk-pk)	
Description:	**peak to peak amplitude of the generated sine wave**
Range:	**0.0 to 5.0**
Default value:	**1.0**
Unit:	**V**

Q	
Description:	**pole quality factor, Q (of the low pass filter component)**
Range:	**1.0 to 20.0**
Default value:	**2.0**
Unit:	**none**

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{f_c}{4}\frac{C_{O2}}{C_{O1}}$$

$$f_0 = \frac{C_2}{C_A}\frac{f_c}{2\pi} = \frac{C_3}{C_B}\frac{f_c}{2\pi}$$

$$Q = \frac{C_A}{C_4}$$

$$\frac{A}{5Q} = \frac{C_1}{C_2}$$

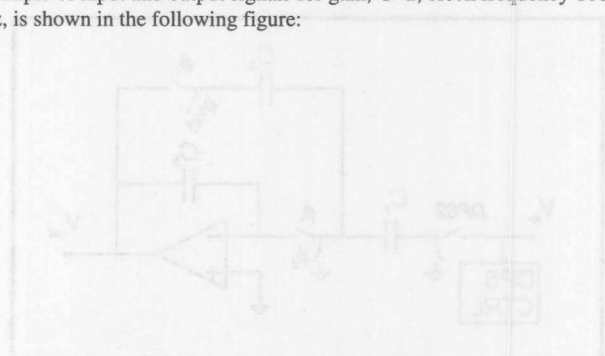
where f_c is the clock frequency and f_o is the output frequency.

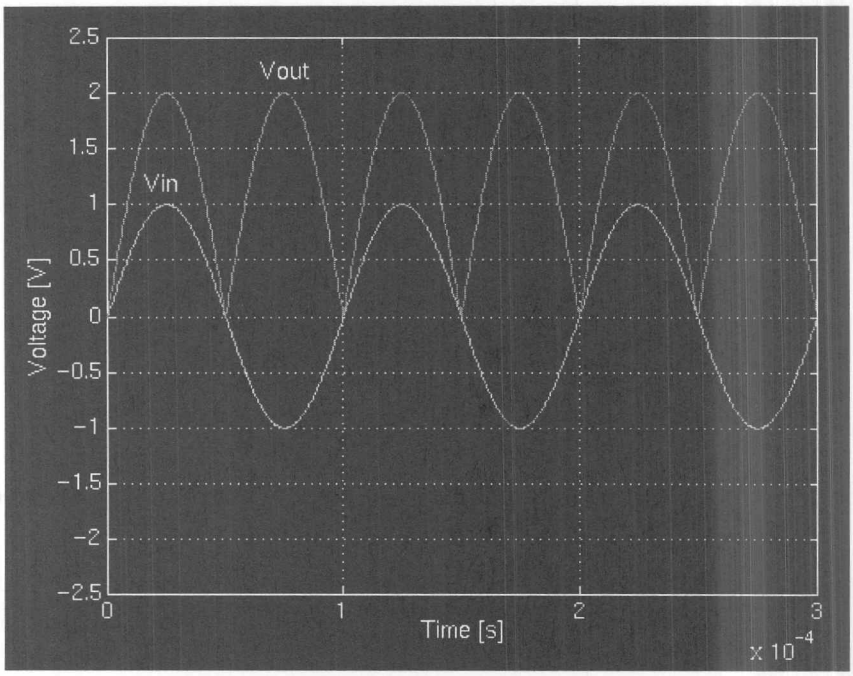
Rectifiers

(R1) Full wave RECTIFIER (with low-pass filter)

This macro is a full wave rectifier (with positive output voltage). It may realize also a single pole low pass filter (LPF) which works as if a one pole LPF was connected in series with the rectifier. The default parameter values disable this feature by moving the corner frequency to the high values.

An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:

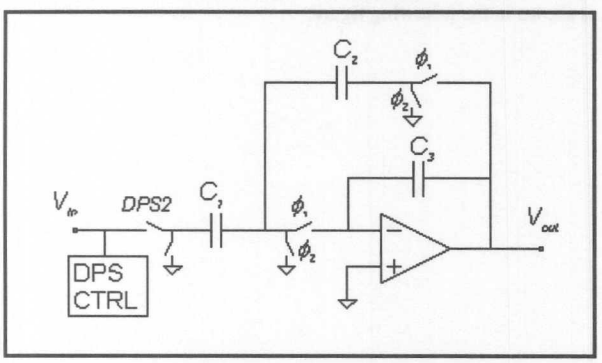




The transfer function of this circuit (working as rectifier) is

$$V_{out} = |V_{in}| \frac{C_1}{C_2} = |V_{in}| G$$

where C_1 and C_2 are the input and feedback capacitors as depicted in the figure:



The symbols ϕ_1, ϕ_2, $DPS2$ specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin<0	if Vin>0
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
Description:	corner frequency, f_0 (frequency at which the gain is $-3+20\log G$ dB, where G is the pass band gain)
Range:	0.62 to 159 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	159.0 - the default value virtually disables the filtering feature
Unit:	kHz

Pass Band Gain	
Description:	voltage gain of the rectifier, G (DC gain of the filter)
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

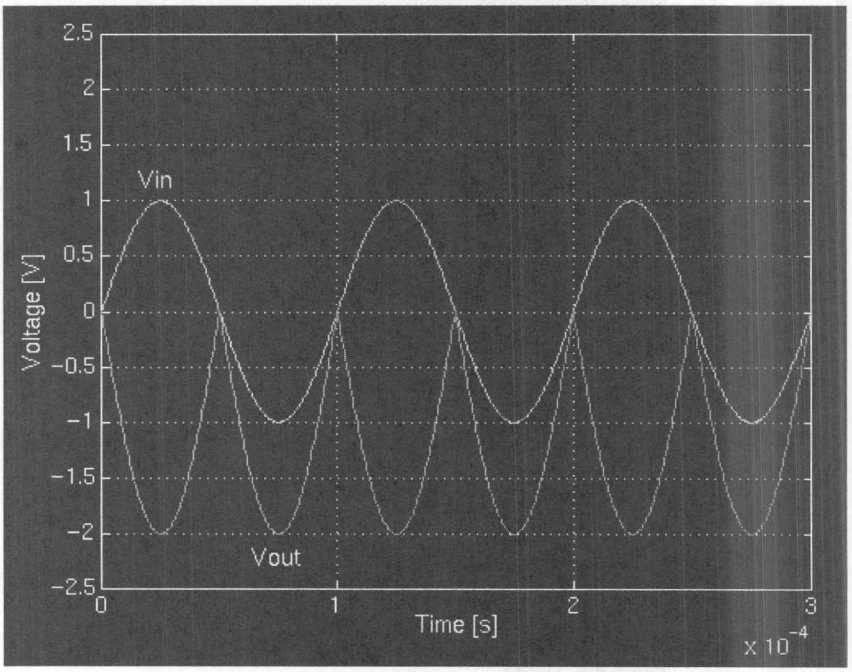
$$f_0 = \frac{C_2}{C_3}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_2}$$

(R2) Inverting full wave RECTIFIER (with low-pass filter)

This macro is a full wave rectifier (with negative output voltage). It may realize also a single pole low pass filter (LPF) which works as if a one pole LPF was connected in series with the rectifier. The default parameter values disable this feature by moving the corner frequency to the high values.

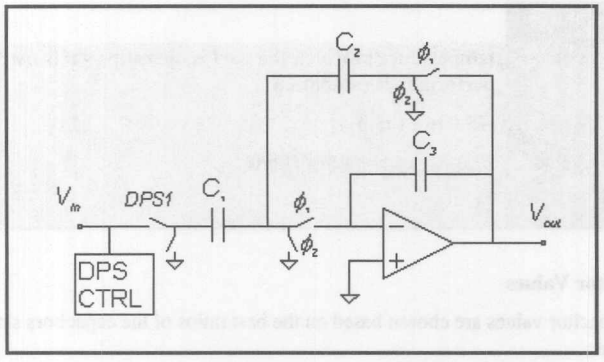
An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit (working as rectifier) is

$$V_{out} = -|V_{in}|\frac{C_1}{C_2} = -|V_{in}|G$$

where C_1 and C_2 are the input and feedback capacitors as depicted in the figure:



The symbols ϕ_1, ϕ_2, $DPS1$ specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	if Vin<0
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
Description:	corner frequency, f_0 (frequency at which the gain is $-3+20\log G$ dB, where G is the pass band gain)
Range:	0.62 to 159 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)
Default value:	159.0 - the default value virtually disables the filtering feature
Unit:	kHz

Pass Band Gain	
Description:	voltage gain of the rectifier, G (DC gain of the filter)
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

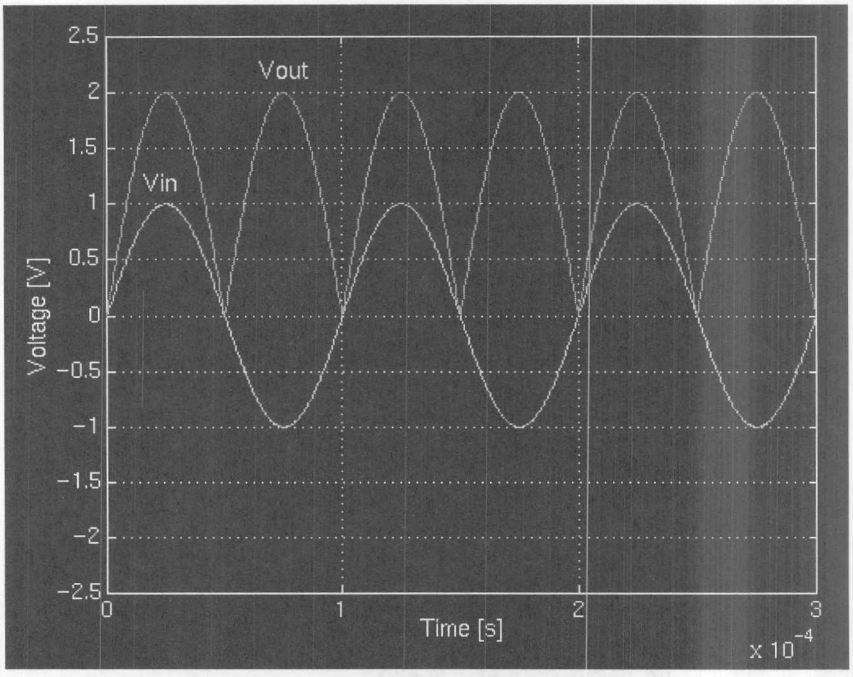
$$f_0 = \frac{C_2}{C_3}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_2}$$

(R3) Full wave RECTIFIER (half cycle with offset comp.)

This macro is a full wave rectifier with offset compensation in one of the two phases.

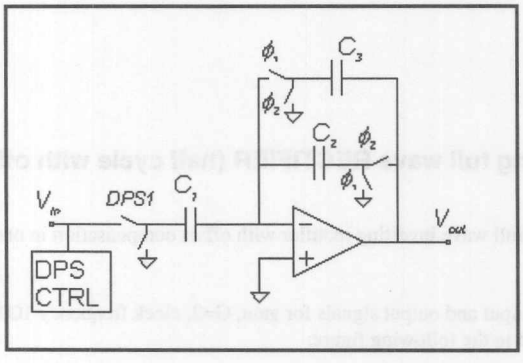
An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit is

$$V_{out} = |V_{in}| \frac{C_1}{C_2} = |V_{in}| G$$

where C_1 and C_2 are the input and feedback capacitors as depicted in the figure:



The symbols ϕ_1, ϕ_2, $DPS1$ specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin<0	if Vin>0
Output valid:	no (Vout=0)	yes, offset compensated

Programmable parameters

Gain	
Description:	voltage gain of the rectifier, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Capacitor Values

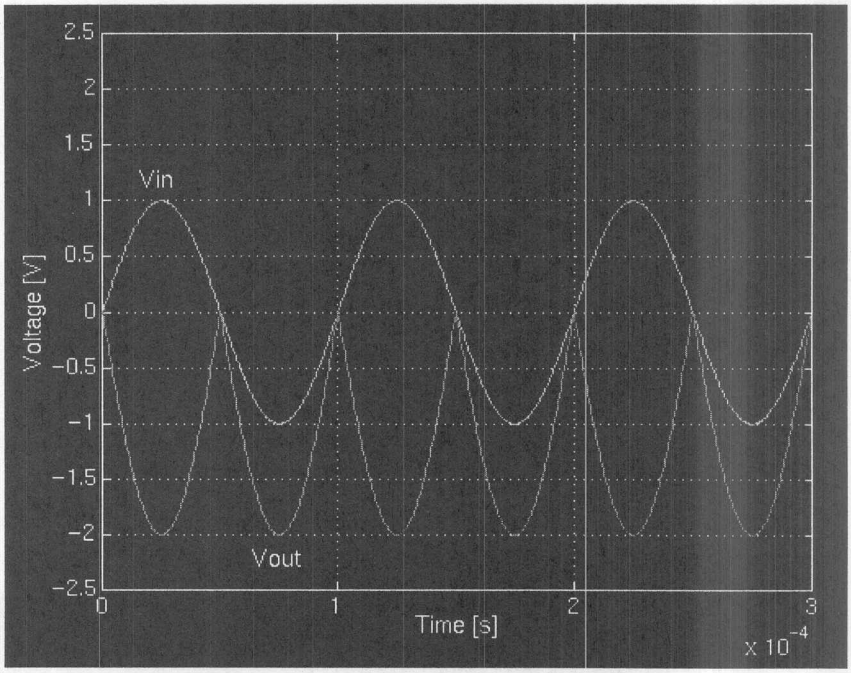
The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$G = \frac{C_1}{C_2}$$

(R4) Inverting full wave RECTIFIER (half cycle with offset comp.)

This macro is a full wave inverting rectifier with offset compensation in one of the two phases.

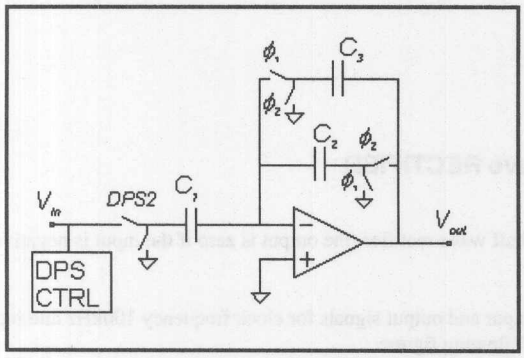
An example of input and output signals for gain, G=2, clock frequency 100kHz, input frequency 10kHz, is shown in the following figure:



The transfer function of this circuit is

$$V_{out} = -|V_{in}|\frac{C_1}{C_2} = -|V_{in}|G$$

where C_1 and C_2 are the input and feedback capacitors as depicted in the figure:



The symbols ϕ_1, ϕ_2, $DPS2$ specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	if Vin<0
Output valid:	no (Vout=0)	yes, offset compensated

Programmable parameters

Gain	
Description:	voltage gain of the rectifier, G
Range:	0.004 to 20.0
Default value:	1.0
Unit:	V/V

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C

Capacitor Values

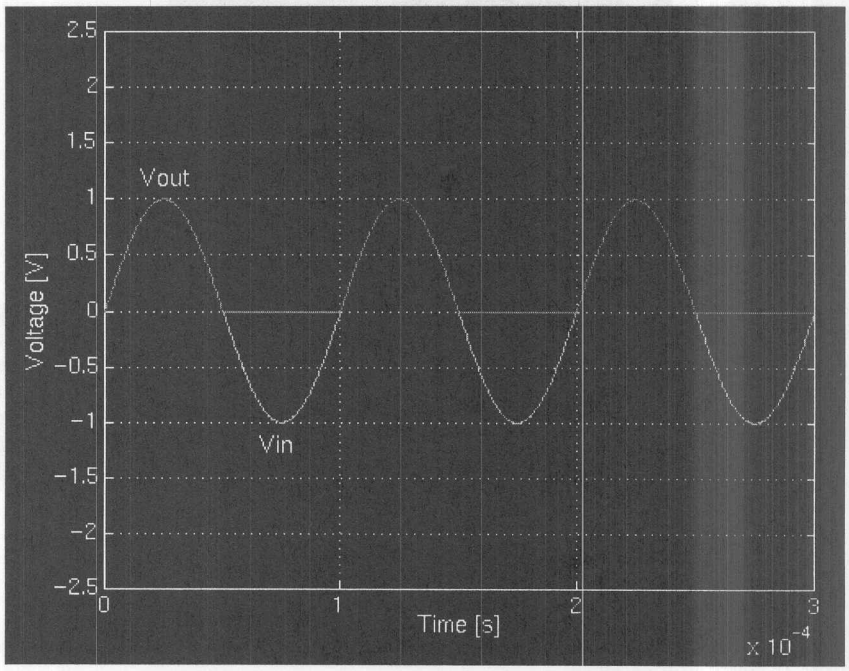
The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$G = \frac{C_1}{C_2}$$

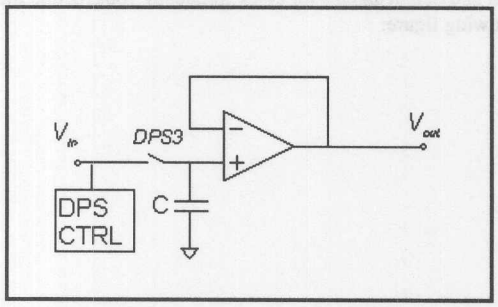
(R5) Half wave RECTIFIER

This macro is a half wave rectifier (the output is zero if the input is negative).

An example of input and output signals for clock frequency 100kHz and input frequency 10kHz, is shown in the following figure:



The schematic of this circuit is presented in the following figure:



The symbol *DPS3* specifies the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	no sampling
Output valid:	yes	yes

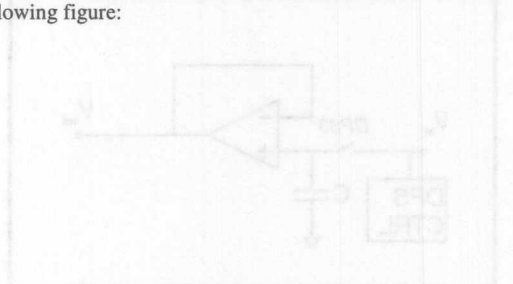
Programmable parameters

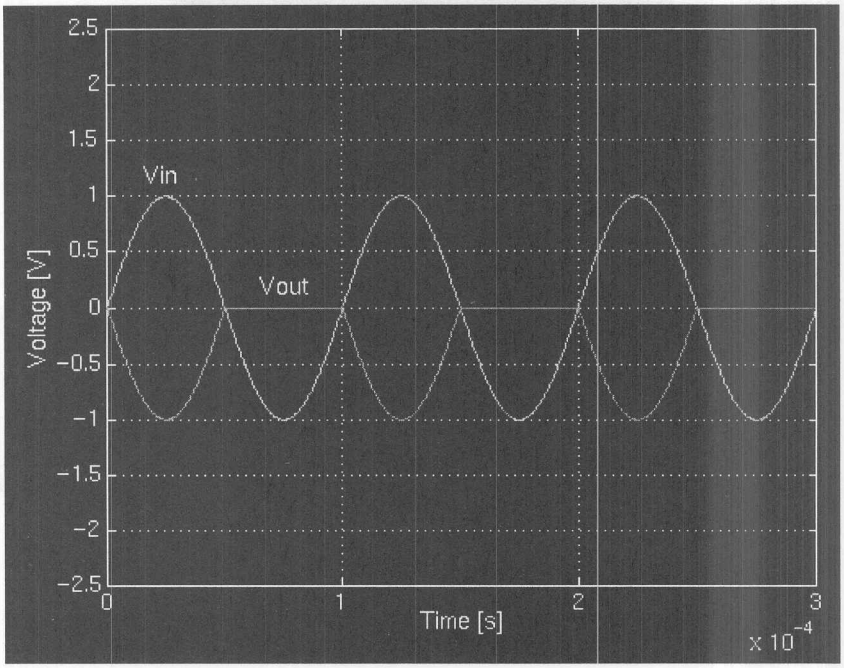
Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

(R6) Inverting half wave RECTIFIER

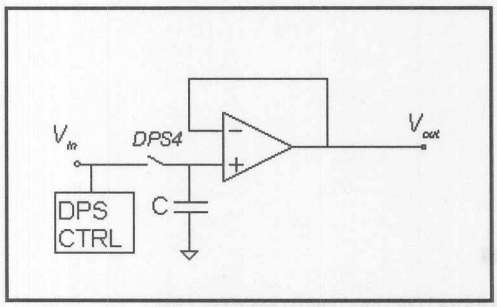
This macro is a half wave rectifier with inverted (negative) output (the output is zero if the input is negative).

An example of input and output signals for clock frequency 100kHz and input frequency 10kHz, is shown in the following figure:





The schematic of this circuit is presented in the following figure:



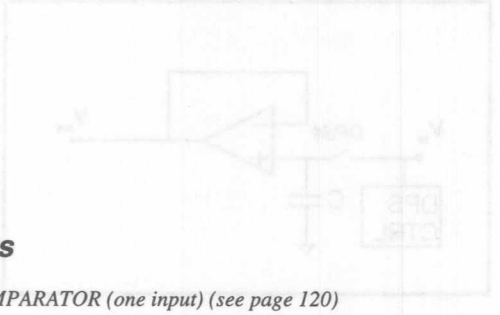
The symbol *DPS4* specifies the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin<0	no sampling
Output valid:	yes	yes

Programmable parameters

Temperature	
Description:	temperature in which the part is operating - it is used only for performance estimation
Range:	-45.0 to +105.0
Default value:	27.0
Unit:	°C



Comparators

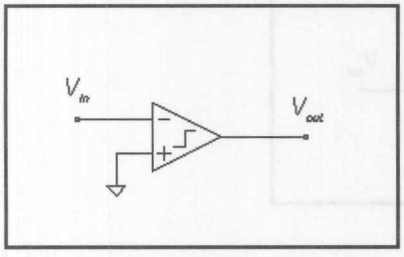
(C1) Inverting COMPARATOR (one input)

This macro is a full cycle inverting comparator.

It has one input which is connected to the negative terminal of a comparator, while the positive terminal is connected to the signal reference (signal 0.0V). If the input is greater than zero, the output is -2.5V, otherwise it is +2.5V.

$$V_{out} = \begin{cases} +2.5V & for \quad V_{in} < 0 \\ -2.5V & for \quad V_{in} > 0 \end{cases}$$

The macro utilizes the built-in comparator circuit available in each cell of the FPPA,



Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**`C**

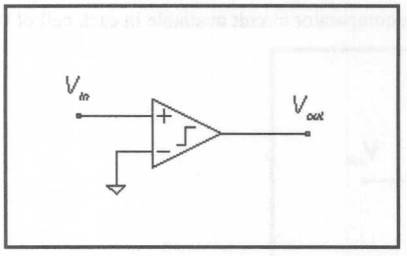
(C2) Non-inverting COMPARATOR (one input)

This macro is a full cycle comparator.

It has one input which is connected to the positive terminal of a comparator, while the positive terminal is connected to the signal reference (signal 0.0V). If the input is greater than zero, the output is +2.5V, otherwise it is -2.5V.

$$V_{out} = \begin{cases} +2.5V & for \quad V_{in} > 0 \\ -2.5V & for \quad V_{in} < 0 \end{cases}$$

The macro utilizes the built-in comparator circuit available in each cell of the FPPA,



Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

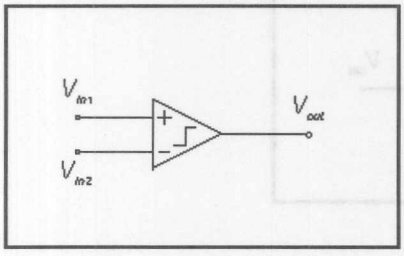
(C3) 2-input COMPARATOR (one cell, may not work in some config.)

This macro is a full cycle comparator. Due to some constraints it may not work in some configurations (for some combinations of inputs). An alternative to this macro is the comparator (C4) which has the disadvantage of occupying two cells while this macro occupies just one cell.

It has two inputs which are compared with each other. If the positive terminal input is greater than the negative input the output is +2.5V, otherwise it is -2.5V,

$$V_{out} = \begin{cases} +2.5V & for \quad V_{in1} > V_{in2} \\ -2.5V & for \quad V_{in1} < V_{in2} \end{cases}$$

The macro utilizes the built-in comparator circuit available in each cell of the FPPA,



Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

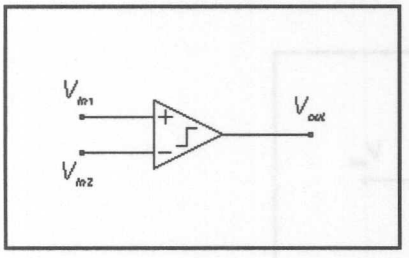
(C4) 2-input COMPARATOR (two cells)

This macro is a full cycle comparator and is an alternative to the smaller macro, (C3), which occupies just one cell (but may not work in some input configurations) while this comparator occupies two cells.

It has two inputs which are compared with each other. If the positive terminal input is greater than the negative input the output is +2.5V, otherwise it is -2.5V,

$$V_{out} = \begin{cases} +2.5V & for \quad V_{in1} > V_{in2} \\ -2.5V & for \quad V_{in1} < V_{in2} \end{cases}$$

The macro utilizes the built-in comparator circuit available in each cell of the FPPA,



Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

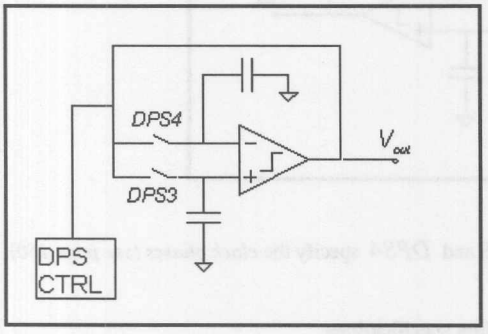
Temperature	
Description:	**temperature in which the part is operating - it is used only for performance estimation**
Range:	**-45.0 to +105.0**
Default value:	**27.0**
Unit:	**°C**

DC Voltage Sources

(V1) DC +2.5V

This macro is a full cycle (output is valid in both phases) positive DC voltage source of +2.5V.

Circuit schematic:



The symbols $DPS3$ and $DPS4$ specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	no
Output valid:	yes	yes

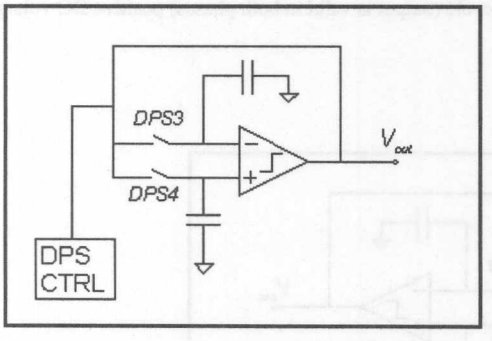
Programmable parameters

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

(V2) DC -2.5V

This macro is a full cycle (output is valid in both phases) negative DC voltage source of -2.5V.

Circuit schematic:



The symbols $DPS3$ and $DPS4$ specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	no
Output valid:	yes	yes

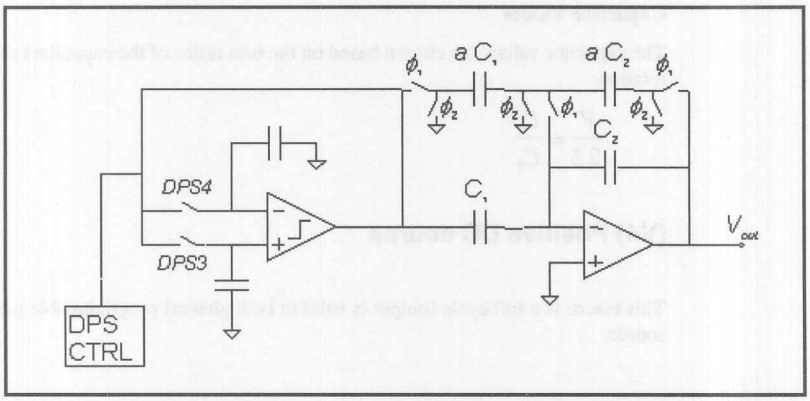
Programmable parameters

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

(V3) Negative DC source

This macro is a full cycle (output is valid in both phases) programmable negative DC voltage source.

Circuit schematic:



The symbols $DPS3$, $DPS4$, ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The constant *a* is not programmable.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	no
Output valid:	yes	yes

Programmable parameters

DC Voltage	
description:	DC voltage level in reference to the signal ground, V
range:	-2.5 to -0.01
default value:	-1.0
unit:	V

Temperature	
description:	temperature in which the part is operating - it is used only for performance estimation
range:	-45.0 to +105.0
default value:	27.0
unit:	°C

Capacitor Values

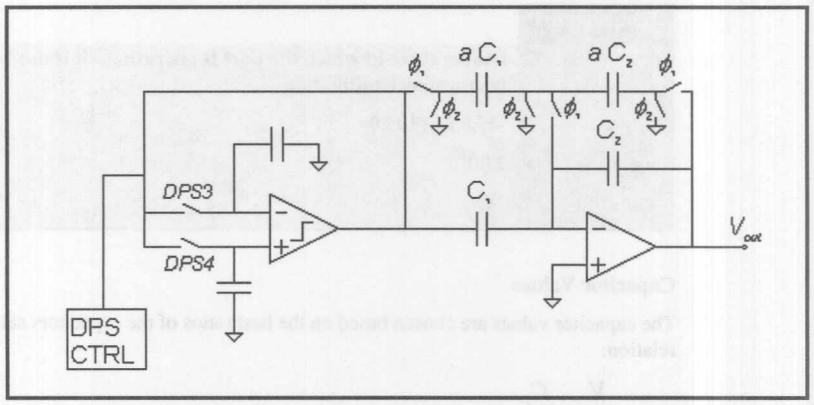
The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$\frac{V}{2.5} = \frac{C_1}{C_2}$$

(V4) Positive DC source

This macro is a full cycle (output is valid in both phases) programmable positive DC voltage source.

Circuit schematic:



The symbols $DPS3$, $DPS4$, ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The constant *a* is not programmable.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	no
Output valid:	yes	yes

Programmable parameters

DC Voltage	
description:	DC voltage level in reference to the signal ground, V
range:	0.01 to 2.5
default value:	1.0
unit:	V

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

Capacitor Values

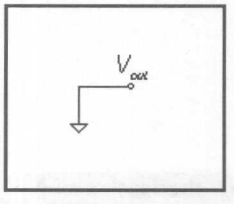
The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$\frac{V}{2.5} = \frac{C_1}{C_2}$$

(V5) DC 0.0V

This macro is a full cycle (output is valid in both phases) DC voltage source of 0.0V (signal ground).

The circuit consists of a connection to the signal ground as shown on the figure:

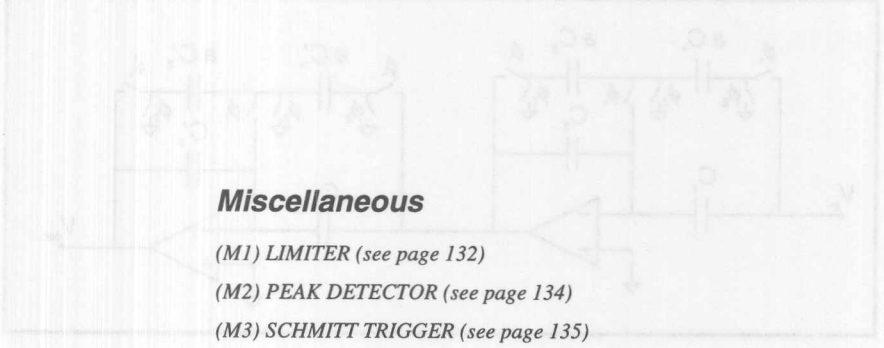


Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	**no**	**no**
Output valid:	**yes**	**yes**

Programmable parameters

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

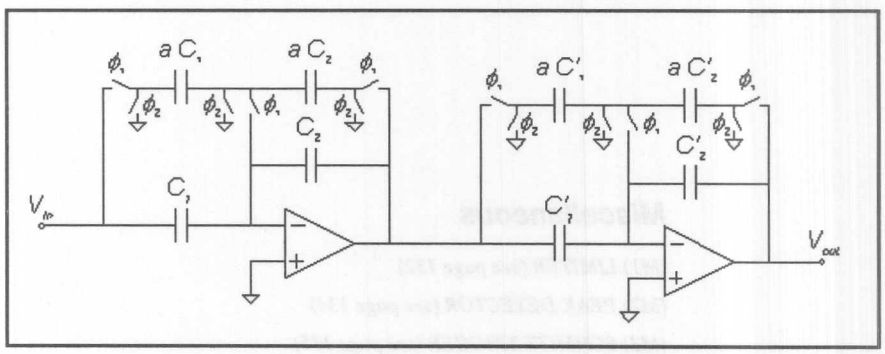


Miscellaneous

(M1) LIMITER

This macro is a full cycle (output is valid in both phases) limiter.

Circuit schematic:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*. The constant *a* is not programmable.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Limiting voltage	
description:	limiting voltage, V_L (the output will be limited to the range $-GV_L$ to $+GV_L$, where G is the limiter's gain)
range:	0.1 to 2.5
default value:	1.0
unit:	V

Gain	
description:	limiter's voltage gain, G
range:	0.004 to 20.0
default value:	1.0
unit:	V/V

Temperature	
description:	temperature in which the part is operating - it is used only for performance estimation
range:	-45.0 to +105.0
default value:	27.0
unit:	°C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

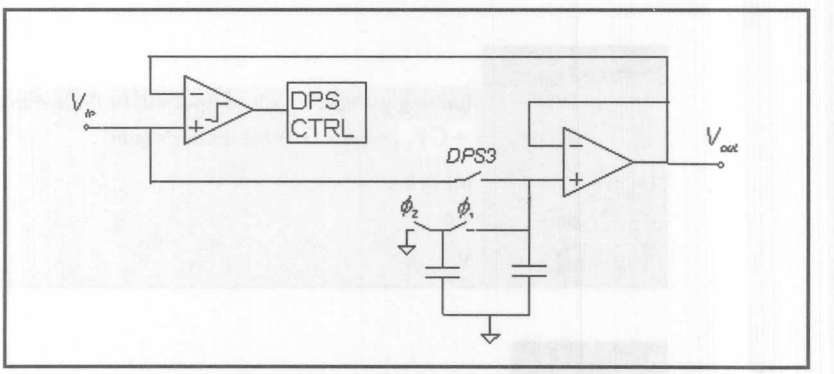
$$\frac{2.5}{V_L} = \frac{C_1}{C_2}$$

$$\frac{GV_L}{2.5} = \frac{C_1'}{C_2'}$$

(M2) PEAK DETECTOR

This macro is a peak detector. It detects a local maximum value of the input voltage.

Circuit schematic:



The symbols ϕ_1, ϕ_2, and $DPS3$ specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

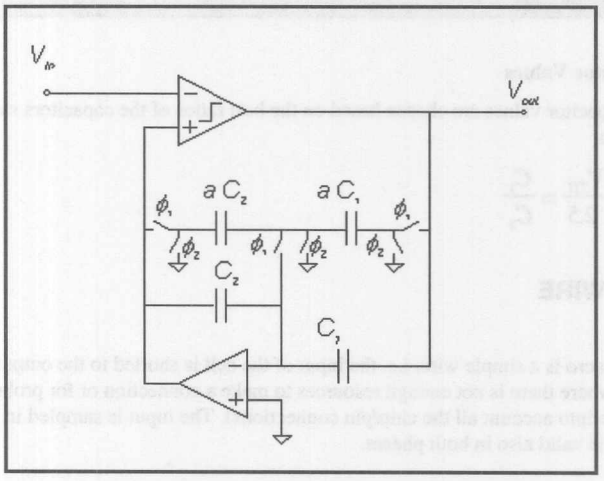
Decay rate	
description:	**decay rate, N_R, measured as number of clock cycles to reduce the output voltage from the peak value down to 50%**
range:	**2.0 to 178**
default value:	**50**
unit:	**clk_cyc**

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

(M3) SCHMITT TRIGGER

This macro is a Schmitt trigger stage (comparator with hysteresis).

The schematic of the circuit is:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	no (Vout=0)	yes (offset compensated)

Programmable parameters

Threshold voltage	
description:	switching threshold voltage, V_{TH}
range:	0.01 to 1.0
default value:	0.5
unit:	V

Temperature	
description:	temperature in which the part is operating - it is used only for performance estimation
range:	-45.0 to +105.0
default value:	27.0
unit:	˚C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$\frac{V_{TH}}{2.5} = \frac{C_1}{C_2}$$

(M4) WIRE

This macro is a simple wire, i.e. the input of the cell is shorted to the output. It may be useful in cases where there is not enough resources to make a connection or for probe calibration purposes (to take into account all the chip/pin connections). The input is sampled in both phases and the output is valid also in both phases.

This macro does not take any parameters and no performance estimates are calculated.

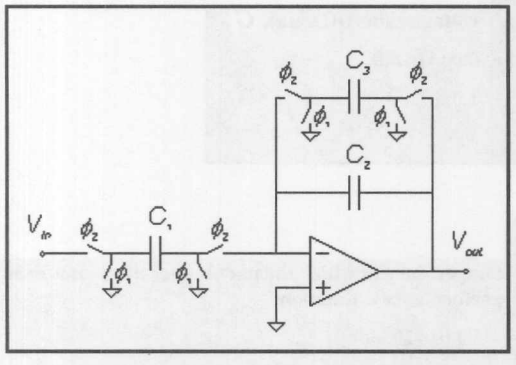
(M5) Inverting damped INTEGRATOR

This macro is a full cycle inverting damped integrator. The design is based on a low pass single pole filter (therefore the input parameters are similar to the ones of the filter).

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{2\pi f_0 G}{s + 2\pi f_0}$$

where G is the pass-band gain and f_0 is the corner frequency. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
description:	**corner frequency, f_0 (frequency at which the gain is $-3+20\log G$ dB, where G is the pass band gain)**
range:	**0.62 to 159 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)**
default value:	**1.0**
unit:	**kHz**

Pass-Band Gain	
description:	**voltage gain (DC gain), G**
range:	**0.004 to 4.0**
default value:	**1.0**
unit:	**V/V**

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**˚C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_3}{C_2}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_3}$$

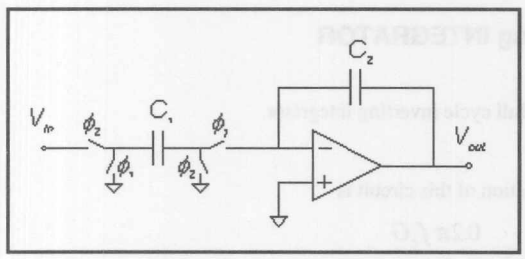
(M6) INTEGRATOR

This macro is a full cycle integrator.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{0.2\pi f_c G}{s}$$

where G is the gain at 1/10 of the clock frequency, f_c. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Gain at fc/10	
description:	voltage gain, G, at the frequency of 1/10 of the clock frequency
range:	0.01 to 20.0
default value:	1.0
unit:	V/V

Temperature	
description:	temperature in which the part is operating - it is used only for performance estimation
range:	-45.0 to +105.0
default value:	27.0
unit:	°C

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$G = \frac{C_1}{0.2\pi C_2}$$

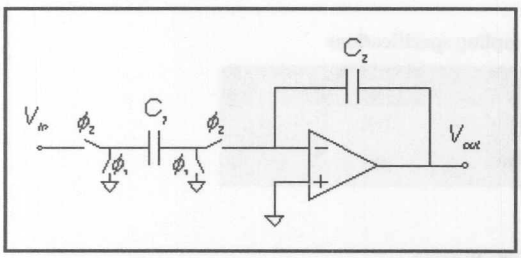
(M7) Inverting INTEGRATOR

This macro is a full cycle inverting integrator.

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{0.2\pi f_c G}{s}$$

where G is the gain at 1/10 of the clock frequency, f_c. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	no	yes
Output valid:	yes	yes

Programmable parameters

Gain at fc/10	
description:	**voltage gain, G, at the frequency of 1/10 of the clock frequency**
range:	**0.01 to 20.0**
default value:	**1.0**
unit:	**V/V**

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$G = \frac{C_1}{0.2\pi C_2}$$

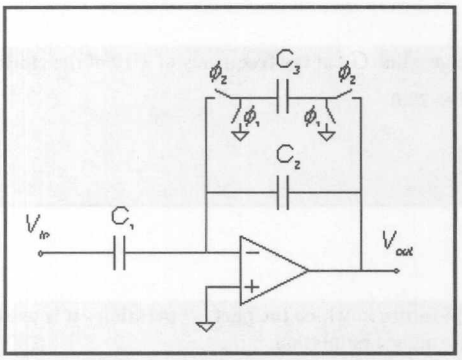
(M8) Inverting DIFFERENTIATOR

This macro is a full cycle inverting differentiator. The designed is based on a high pass single pole filter (therefore the input parameters are similar to the ones of the filter).

The transfer function of this circuit is

$$\frac{V_{out}(s)}{V_{in}(s)} = -\frac{Gs}{s+2\pi f_0}$$

where G is the pass-band gain and f_0 is the corner frequency. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150)*.

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	yes	yes
Output valid:	yes	yes

Programmable parameters

Corner Frequency	
description:	**corner frequency, f_0 (frequency at which the gain is $-3+20\log G$ dB, where G is the pass band gain)**
range:	**0.62 to 159 kHz for the default clock frequency of 1MHz (for different clock frequencies the range has to be divided by the ratio $1000/f_c$ where f_c is the clock frequency in kHz)**
default value:	**50.0**
unit:	**kHz**

Pass-Band Gain	
description:	**voltage gain (DC gain), G**
range:	**0.004 to 4.0**
default value:	**1.0**
unit:	**V/V**

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relations:

$$f_0 = \frac{C_3}{C_2}\frac{f_c}{2\pi}$$

$$G = \frac{C_1}{C_2}$$

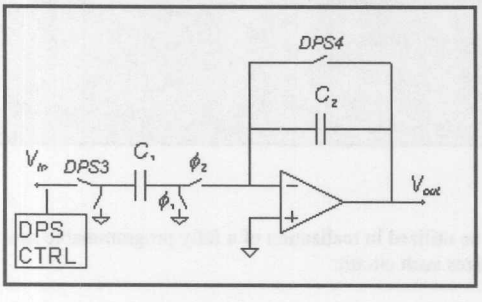
(M9) RAMP

This macro is a full cycle non-inverting integrator with a reset. When the input voltage is positive the circuit performs integration and when the input is negative the integrating capacitor is being discharged (in one cycle). The following are the parameters that may be specified for programming this macro:

The input voltage is assumed to have positive constant value for integration and negative value when the reset operation is requested. The transfer function, assuming that the input is constant and integration is in progress, is

$$\frac{V_{out}}{V_{in}} = n\frac{C_1}{C_2}$$

where n is the number clock cycles since the beginning of the integration period. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 , ϕ_2, $DPS3, DPS4$ specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	no
Output valid:	yes	yes

Programmable parameters

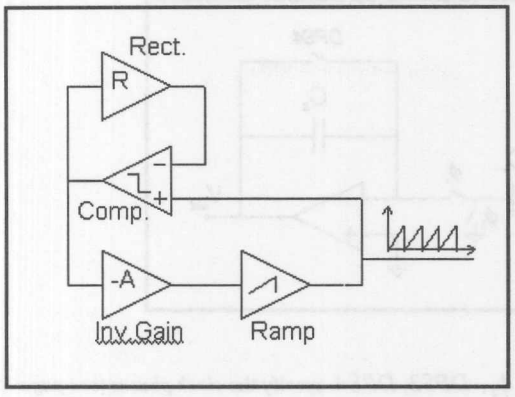
ClkCyc/1V (for 1V Vin)	
description:	**number of clock cycles to achieve 1V change in output when the input voltage is DC 1V**
range:	**1 to 255 (if the specified input voltage is different from 1V than this range should be divided by the value of input voltage)**
default value:	**255**
unit:	**1/V**

Const Vin	
description:	**value of the DC input voltage**
range:	**0.01 to 2.5**
default value:	**1.0**
unit:	**V**

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

Example:

This macro cell may be utilized in realization of a fully programmable saw tooth generator. The following circuit realizes such circuit:



Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$N = \frac{C_2}{C_1}$$

where N is the number of clock cycles to achieve 1V change in the output when 1V constant input is applied.

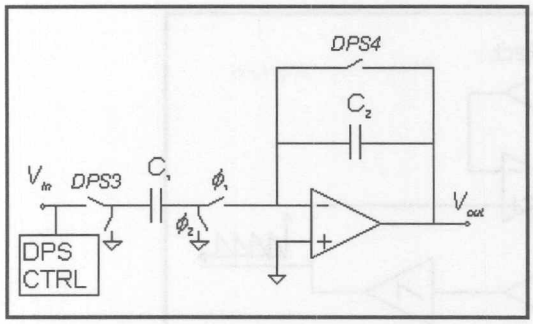
(M10) Negative RAMP

This macro is a full cycle non-inverting integrator with a reset. When the input voltage is positive the circuit performs integration and when the input is negative the integrating capacitor is being discharged (in one cycle). The following are the parameters that may be specified for programming this macro:

The input voltage is assumed to have positive constant value for integration and negative value when the reset operation is requested. The transfer function, assuming that the input is constant and integration is in progress, is

$$\frac{V_{out}}{V_{in}} = -n\frac{C_1}{C_2}$$

where n is the number clock cycles since the beginning of the integration period. The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 , ϕ_2, $DPS3, DPS4$ specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	no
Output valid:	yes	yes

Programmable parameters

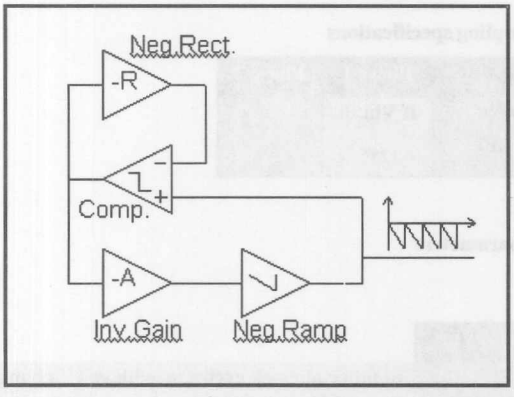
ClkCyc/1V (for 1V Vin)	
description:	number of clock cycles to achieve 1V change in output when the input voltage is DC 1V
range:	1 to 255 (if the specified input voltage is different from 1V than this range should be divided by the value of input voltage)
default value:	255
unit:	1/V

Const Vin	
description:	value of the DC input voltage
range:	0.01 to 2.5
default value:	1.0
unit:	V

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

Example:

This macro cell may be utilized in realization of a fully programmable negative saw tooth generator. The following circuit realizes such circuit:



Capacitor Values

The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$N = \frac{C_2}{C_1}$$

where N is the number of clock cycles to achieve 1V change in the output when 1V constant input is applied.

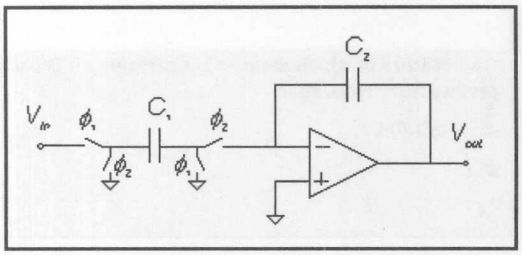
(M11) RAMP (Triangle)

This macro is a full cycle non-inverting integrator implemented to perform as a triangle wave generator assuming that the input is a square wave.

The slope of the output voltage assuming that the input is constant, is

$$\frac{dV_{out}}{dt} = \varepsilon \frac{C_1}{C_2} \quad \text{where} \quad \varepsilon = \begin{cases} +1 & \text{for } V_{in} > 0 \\ -1 & \text{for } V_{in} < 0 \end{cases}$$

The circuit realizing this macro is shown in the following figure:



The symbols ϕ_1 and ϕ_2 specify the *clock phases (see page 150).*

Input-output sampling specifications

	Phase 1	Phase 2
Input sampled:	if Vin>0	no
Output valid:	yes	yes

Programmable parameters

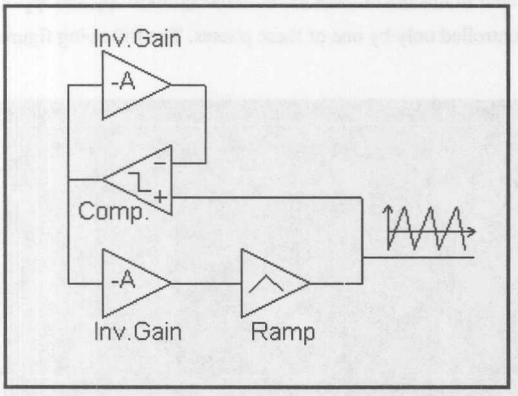
ClkCyc/1V (for 1V Vin)	
description:	number of clock cycles to achieve 1V change in output when the input voltage is DC 1V
range:	1 to 255 (if the specified input voltage is different from 1V than this range should be divided by the value of input voltage)
default value:	255
unit:	1/V

Const Vin	
description:	value of the DC input voltage
range:	0.01 to 2.5
default value:	1.0
unit:	V

Temperature	
description:	**temperature in which the part is operating - it is used only for performance estimation**
range:	**-45.0 to +105.0**
default value:	**27.0**
unit:	**°C**

Example:

This macro cell may be utilized in realization of a fully programmable triangle wave generator. The following circuit realizes such generator:



Capacitor Values

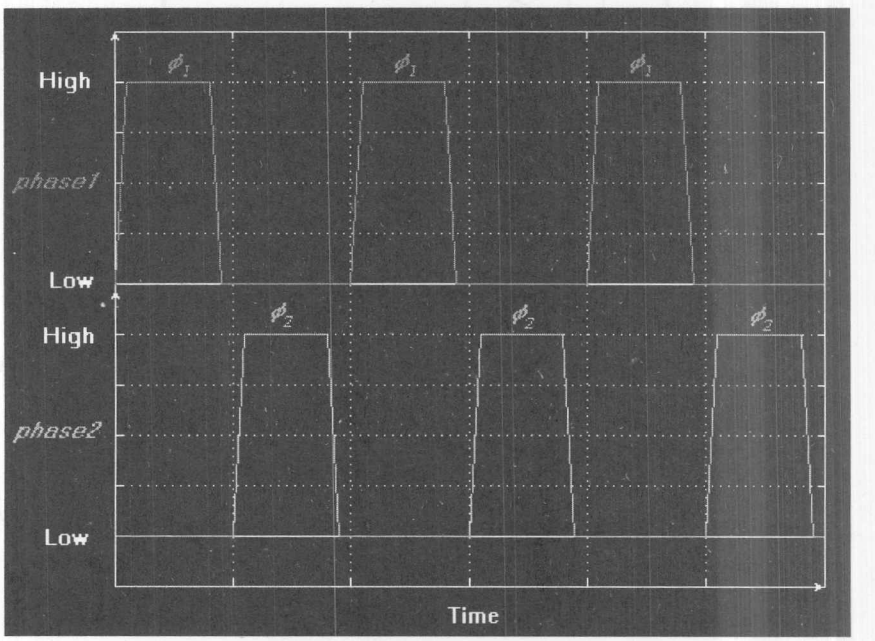
The capacitor values are chosen based on the best ratios of the capacitors satisfying the following relation:

$$N = \frac{C_2}{C_1}$$

where N is the number of clock cycles to achieve 1V change in the output when 1V constant input is applied.

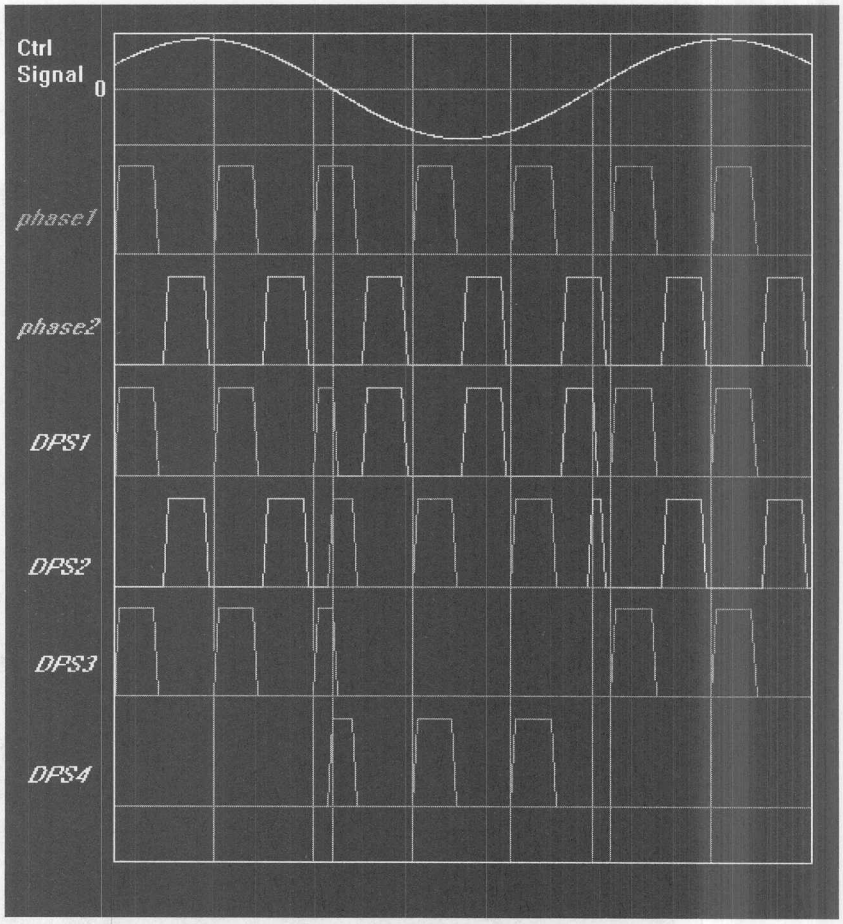
Clock Phases

The switched capacitor circuit is controlled with a two phase clock. The high value of the clock signal closes a switch and the low value opens it. The two phases in which the clock has high value are represented in this documentation with the symbols ϕ_1 and ϕ_2. Each dynamic switch of the circuit is controlled only by one of these phases. The following figure shows the waveforms of ϕ_1 and ϕ_2:



Note that the phases do not overlap each other (there is a non-overlapping period in which both phases have low value).

Some circuits may require swapping or stopping the controlling clock phases of some switches. This is referred to as Dynamic Phase Swapping or stopping (DPS). This dynamic phase swapping or stopping is represented with special phase names as shown below.



The signals *DPS1* to *DPS4* have a transition point in which the controlling signal (top of the above figure) changes signs. Note tat when the controlling signal reaches the value of zero (0) the phase is instantly swapped (signal goes to zero) without waiting for the current phase to terminate. There are four DPS types as depicted on the above figure:

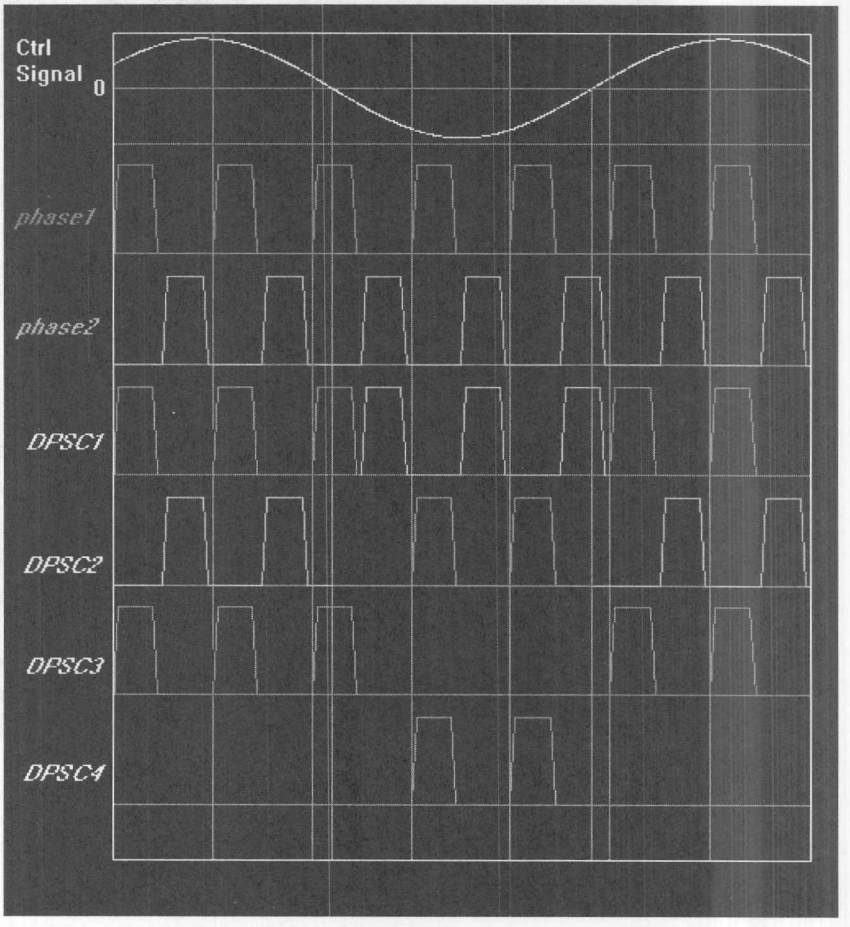
DPS1 - when the controlling signal has a positive value the clock is in phase1 and when the controlling signal has a negative value the clock is in phase2.

DPS2 - when the controlling signal has a positive value the clock is in phase2 and when the controlling signal has a negative value the clock is in phase1.

DPS3 - when the controlling signal has a positive value the clock is in phase1 and when the controlling signal has a negative value the clock has a low value (the controlled switch is open).

DPS4 - when the controlling signal has a negative value the clock is in phase1 and when the controlling signal has a positive value the clock has a low value (the controlled switch is open).

Some circuits may require that the active phase is not interrupted when the controlling signal reaches zero as shown above. In those cases the following set of DPS types is applied in which the phase swapping is synchronized with the phase 1.



The signals DPSC1 to DPSC4 are synchronized with the controlling signal (Ctrl Signal) in the same way as the DPS types DPS1 to DPS4.

Index